

AD A101357

(12) LEVEL III

44
AD

AD-E400 645

TECHNICAL REPORT ARLCD-TR-80041

**AN ANTI-RADIATION PROJECTILE (ARP) TERMINAL
EFFECTS SIMULATION COMPUTER PROGRAM (ARPSIM)**

R. D. WEBSTER

DTIC
ELECTED
JUL 13 1981
S D
B

JUNE 1981



US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
LARGE CALIBER
WEAPON SYSTEMS LABORATORY
DOVER, NEW JERSEY

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

FILE COPY
EFC

81 6 29 024

The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

Destroy this report when no longer needed. Do not return it to the originator.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER TECHNICAL REPORT ARLCD-TR-80041	2. GOVT ACCESSION NO. <i>A D - A 104 357</i>	3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) AN ANTI-RADIATION PROJECTILE (ARP) TERMINAL EFFECTS SIMULATION COMPUTER PROGRAM (ARPSIM)	5. TYPE OF REPORT & PERIOD COVERED Final		
7. AUTHOR(s) R. D. Webster	6. PERFORMING ORG. REPORT NUMBER		
9. PERFORMING ORGANIZATION NAME AND ADDRESS ARRADCOM, LCWSL Systems Development and Modelling Div (DRDAR-LCS) Dover, NJ 07801	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS		
11. CONTROLLING OFFICE NAME AND ADDRESS ARRADCOM, TSD STINFO Div (DRDAR-TSS) Dover, NJ 07801	12. REPORT DATE June 1981		
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. NUMBER OF PAGES 105		
16. DISTRIBUTION STATEMENT (of this Report)	15. SECURITY CLASS. (of this report) Unclassified		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)	18a. DECLASSIFICATION/DOWNGRADING SCHEDULE		
18. SUPPLEMENTARY NOTES			
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)			
Guidance accuracy	Direct hit effect		
Fuzing schemes	Radar blast		
Homing projectile	Vehicle body blast		
Monte Carlo simulation	Linear trajectories		
Warhead fragmentation effects	Radar emitting target		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)			
This report is the documentation for a computer code developed primarily to aid development engineers by providing estimates of the relative importance of components in terms of effectiveness.			
<p>The ARPSIM computer model was developed in support of a requirement to estimate the effectiveness of the various kill mechanisms (fragmentation, antenna blast, vehicle blast, and direct hit) of an Anti-Radiation Projectile (ARP) against a typical air defense radar-emitting target. A Monte Carlo technique</p>			

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 68 IS OBSOLETE

(cont)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

is used to generate estimates of the probability of kill for a single ARP fired against a single target. The influence of various fuzing schemes and guidance errors are considered in determining burst points.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

CONTENTS

	Page
Introduction	1
Program Flow	1
Terminal Effects	3
Coordinate System	3
Attack Angle	4
Guidance Errors	4
Fuzing	6
Target	8
Burst Point	8
Direct Hit	10
Blast	10
Fragmentation	10
Monte Carlo Estimates	13
Conclusions	13
Recommendations	14
References	15
Appendices	
A User Guide	17
B Example	27
C FORTRAN Listing	37
Distribution List	103

S DTIC
 ELECTE **D**
 JUL 13 1981
B

Accession For	
<input checked="" type="checkbox"/> NTIS GRA&I <input type="checkbox"/> DTIC TAB <input type="checkbox"/> Unannounced <input type="checkbox"/> Justification	
By _____	
Distribution/ _____	
Availability Codes	
Dist	Avail and/or Special
A	

FIGURES

	Page
1 Program flow	2
2 Coordinate system	4
3 Attack angle	4
4 Guidance errors	5
5 Fuzing angle	7
6 Linear fuzing	7
7 Target description	9
8 Blast radii vs height	11
9 Blast kill probability vs height	11
10 Radar blast function	11
11 Fragmentation grid interpolation	12

INTRODUCTION

ARPSIM is a computer program developed to provide estimates of the terminal effectiveness of an Anti-Radiation Projectile (ARP) fired against an air defense, radar-emitting target.

The primary objective of ARPSIM is to provide the user with a tool to parametrically ascertain the sensitivity of the ARP to warhead, guidance, and fusing design changes.

The ARPSIM model simulates single round terminal conditions from the time when the ARP is flying a straight line trajectory at some fixed attack elevation in the vicinity of the target. Trajectories are determined from guidance errors distributed about a specified homing point. No further trajectory alterations are made. Fuzing points on the target are specified, and when fuzing conditions are satisfied, a burst point is established along the selected trajectory. The proximity of the burst point to the target determines the magnitude of kill probabilities for blast, direct hit, and fragmentation effects. Separate blast kills for both the target body and radar antenna can be estimated. Fragmentation effects are based upon terminal effectiveness estimates generated by the full spray material lethal area (MAE) computer code (refs 1 and 2).

The ARPSIM program is coded in FORTRAN for interactive use on the CDC 6500/6600 in the INTERCOM mode. The user is prompted for data entry. Also, at the option of the user, an inpvc guide can be generated prior to each use. Fragmentation effects are estimated from data previously generated by the MAE program relative to conditional kill probabilities. Optionally, a function, $P_k(r)$, can be provided to specify fragmentation kill probability as a function of range. Comments are added throughout the FORTRAN code for better understanding and for development of future options for the code.

A user guide, an example of a computer run, and a FORTRAN code listing are presented as appendixes A, B, and C.

PROGRAM FLOW

For each Monte Carlo sample, a simulation of the terminal characteristics of the ARP is made beginning at a time prior to fuzing during the ARP flight after final corrections to the trajectory have been made and when the remaining trajectory is linear at a fixed attack angle. The sequence of events for each simulation is:

1. An attack angle is chosen which provides a straight line flight path with respect to a specified homing point.
2. A trajectory is chosen based upon the guidance errors with respect to the homing point.
3. A fuzing point along the chosen flight path is determined.

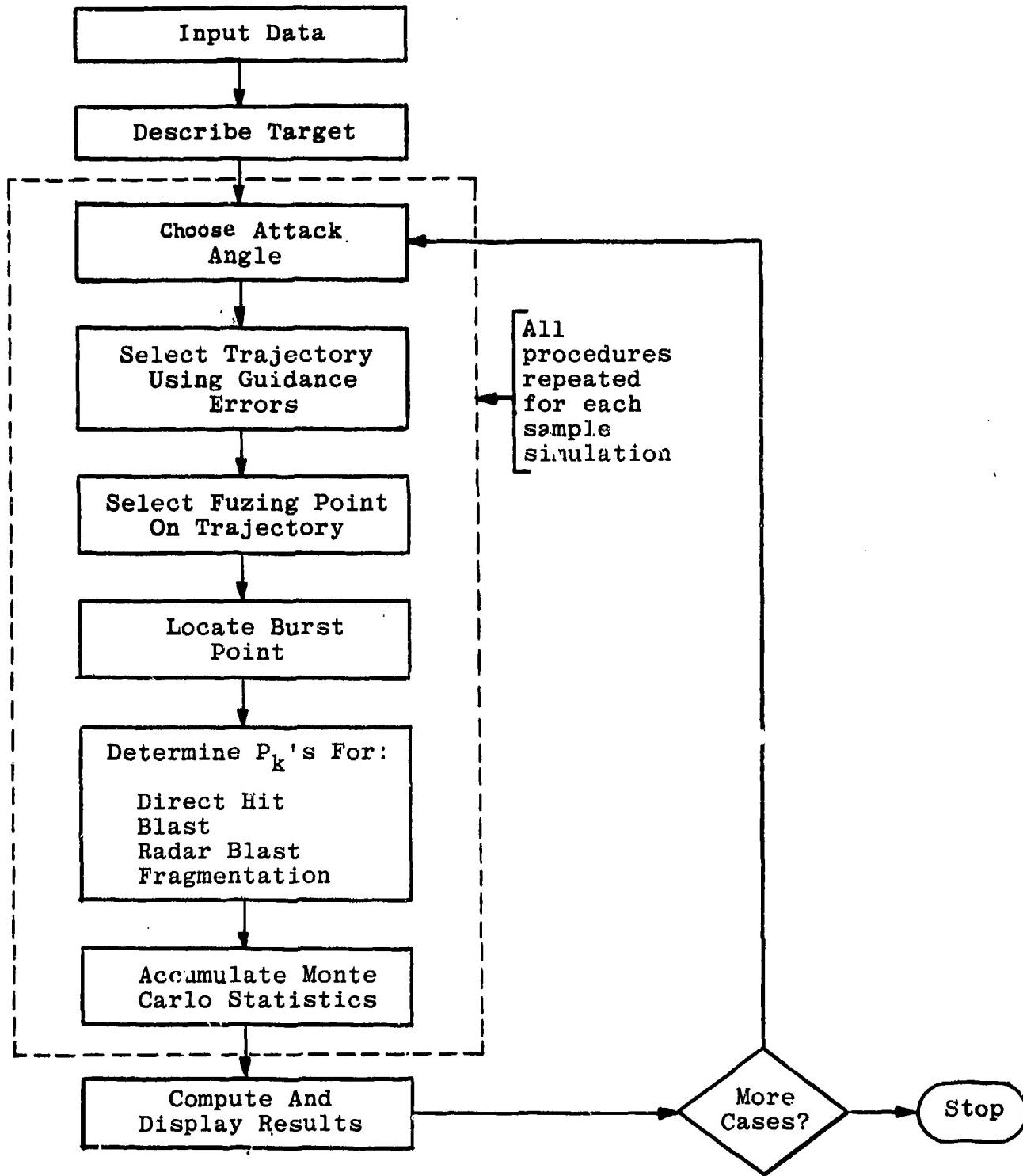


Figure 1. Program flow

4. A burst point is established based on the type of fuze, direct hit potential and possible backup fuzing or ground impact prior to nominal fuzing.

The proximity of the burst point to the target yields estimates of kill probability for direct hit, target body blast, radar blast, and fragmentation effects. The overall kill probability for each simulation is determined from the individual kill mechanism effects. This process is repeated for each simulation to provide the desired estimates of ARP terminal effectiveness. The above-described program flow is illustrated in figure 1.

The following subsections briefly describe portions of the model in the approximate order in which they follow the program flow.

Terminal Effects

Terminal effects are measured in terms of direct hit, blast, and fragmentation. Knowledge of the ARP warhead characteristics as well as the target's vulnerability to each of these effects is essential. Consequently, a preliminary analysis is required of the vulnerability of the target to the ARP warhead. Fragmentation effects are provided in either of two distinct formats: a P_k grid which yields conditional kill probability as a function of burst point proximity to the target, or a P_k vs R (range) function which provides the kill probability data as a function of range only; i.e., azimuth characteristics are averaged for each range from projectile burst to target. These functions are provided by the MAE program. Direct hit and blast effects are estimated from standard target vulnerability analysis.

The overall kill probability for each Monte Carlo sample is based upon these individual effects and is computed as:

$$P_k = 1 - (1 - P_{DH})(1 - P_{RDR})(1 - P_{BLST})(1 - P_F)$$

where

P_{DH} = direct hit kill probability,

P_{RDR} = radar blast kill probability,

P_{BLST} = vehicle blast kill probability,

and

P_F = fragmentation kill probability.

Coordinate System

The simulation uses a rectangular coordinate system whose origin is at ground zero of the target center of vulnerability. Target heading establishes the negative range direction (-R); positive deflection (D) is to the left (driver's side) of the target; height (H) is measured from the ground (positive up) (fig. 2).

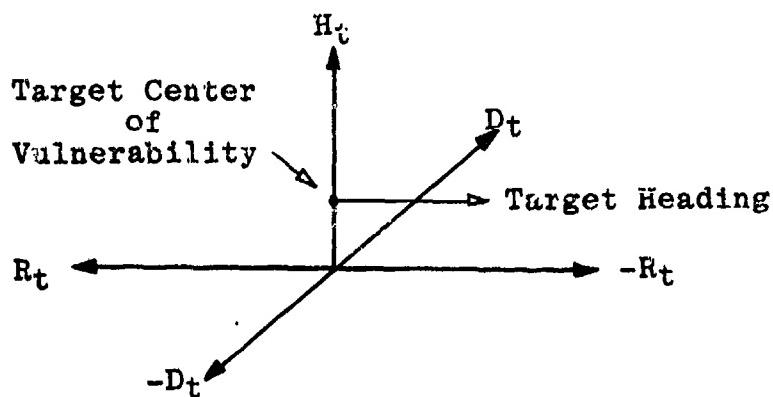


Figure 2. Coordinate system

Attack Angle

The attack angle is the combination of both elevation and azimuth angles which define the direction of the incoming ARP with respect to the coordinate system for the target. Azimuth is measured from the negative range direction toward the positive deflection. The elevation angle, ω , is measured from the horizontal in the positive height direction (fig. 3). Azimuth can be either fixed or chosen randomly for each simulation. Elevation is chosen from a Gaussian distribution with a specified mean, μ_ω , and standard deviation, σ_ω . The attack angle orients the direction of the ARP flight path (trajectory).

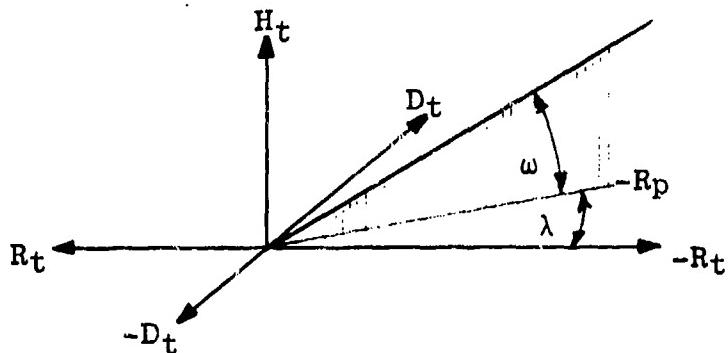


Figure 3. Attack angle

Guidance Errors

Guidance errors are Gaussian and are specified by either the standard deviations in deflection and height or CEP in deflection and height. These errors are defined in the plane normal to the ARP trajectory and passing through the homing point. The location of the guidance plane and the selection of a sample trajectory through the point (GR, GD, GH) are illustrated in figure 4. The determination of the point (GR, GD, GH) is as follows:

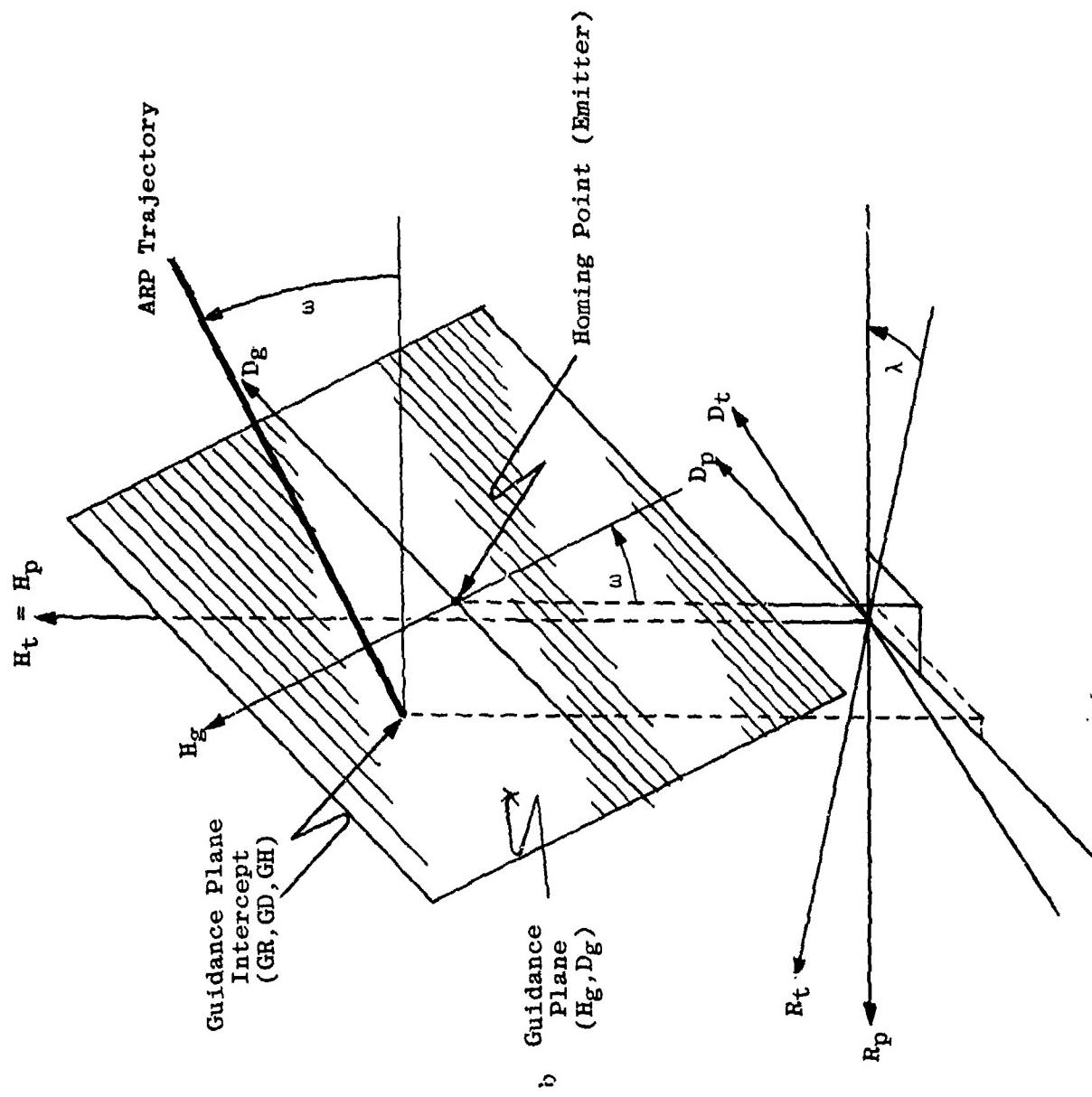


Figure 4. Guidance errors

First, the homing point (GMR, GMD, GMH), defined in the target coordinate system (R_t , D_t , H_t), is rotated through the azimuth angle, λ .

$$\begin{aligned} GMR' &= GMR \cos(\lambda) - GMD \sin(\lambda) \\ GMD' &= GMD \cos(\lambda) + GMR \sin(\lambda) \end{aligned}$$

Then GR, GD, and GH are defined based on the sampled errors about the rotated homing point. Then

where H, D are random normal deviates with $\mu = 0$, $\sigma = 1$,

$$\begin{aligned} GR &= GMR' + H \cdot \sigma_h \cdot \sin(\omega) \\ GD &= GMD' + D \cdot \sigma_d \\ \text{and, } GH &= GMH + H \cdot \sigma_h \cdot \cos(\omega) \end{aligned}$$

where GR, GD, GH are in the R_p , D_p , H_p (projectile) coordinate system and σ_h , σ_d are the standard deviations in height and deflection, respectively, of the guidance error in the guidance plane (H_g , D_g).

Fuzing

Six options are available for primary fuzing; both point detonating (PD) and proximity (VT) backup fuzes can be considered. Each of the primary fuzes is described below:

Gaussian Fuzing Angle

Fuze glitter points are specified on the target and a single glitter point is selected as either the first glitter point encountered or, optionally, chosen randomly for each simulation. When the angle between the flight path and a line from the ARP to the glitter point is equal to the fuzing angle, ϕ , the point on the trajectory at the vertex of the angle is taken to be the fuzing point (fig. 5). The fuze angle for each simulation is selected from a Gaussian distribution as,

$$\phi = \mu_\phi + v \cdot \sigma_\phi$$

where v is a random normal deviate with $\mu = 0$ and $\sigma = 1$.

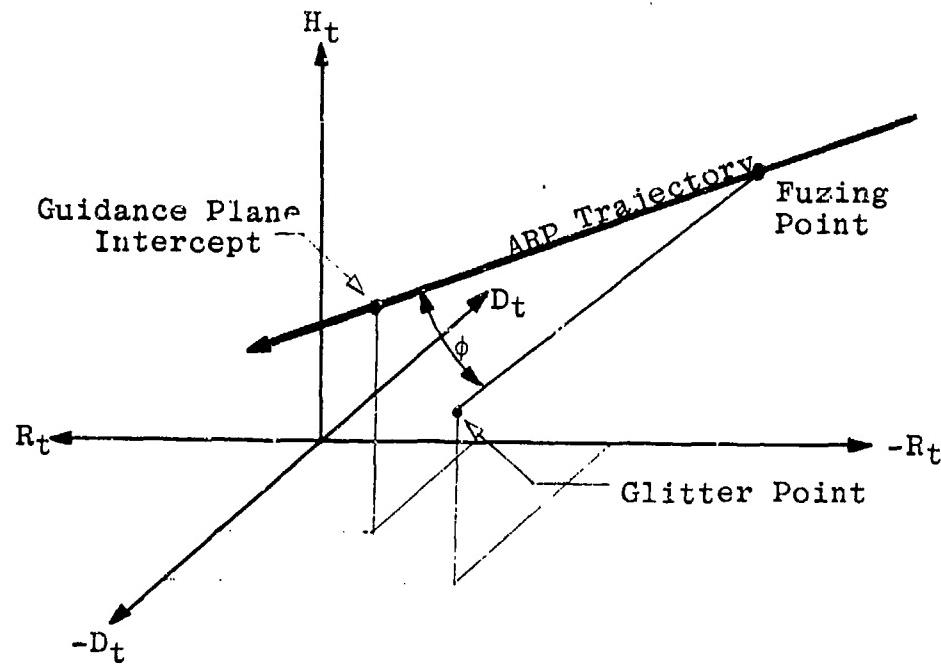


Figure 5. Fuzing angle

Uniform Fuzing Angle

Identical to the Gaussian fuzing angle except that ϕ is chosen as uniformly random between specified limits for each simulation.

Linear Fuzing

Fuzing occurs at some distance along the ARP flight path measured from the guidance plane. The distance along the flight path is chosen from a Gaussian distribution with a specified mean, μ_1 and standard deviation, σ_1 (fig. 6). Given the ARP terminal velocity, linear fuzing can be used to represent a time fuze where time is measured from the guidance plane. If μ_t , σ_t represent the Gaussian parameters for a time fuze, then where V_T is the ARP terminal velocity, $\mu_1 = V_T * \mu_t$ and $\sigma_1 = V_T * \sigma_t$.

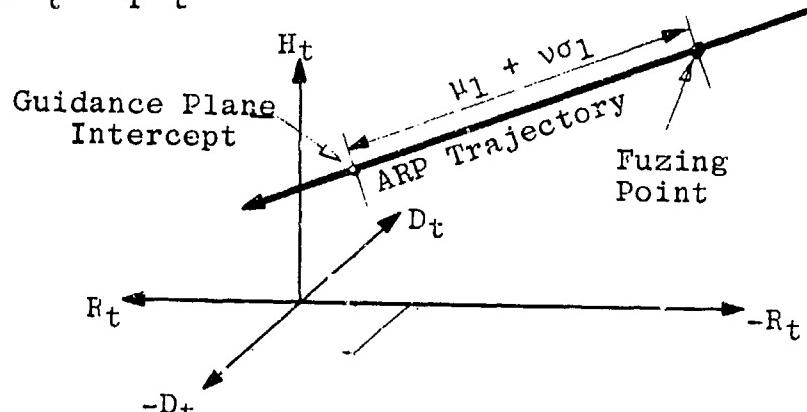


Figure 6. Linear fuzing

Height Fuzing

Fuzing occurs at a specific height above the ground. Height is chosen from a Gaussian distribution where the mean and standard deviation are specified. The point on the ARP flight path which corresponds to the selected height is the fuzing point.

VT Fuze

A VT fuze functioning distribution is considered by specifying the cumulative distribution function of fuzing height. A fuzing height is chosen according to sampling from that distribution and the fuzing point is the point on the ARP flight path which corresponds to the selected height.

PD Fuze

The intersection of the flight path with the ground establishes the PD fuzing point.

All of the above described primary fuze options can have either a PD or VT backup fuze. The backup fuze is used if a test for primary fuze functioning fails; otherwise, the primary fuze establishes the fuzing point unless a VT backup fuze point occurs at a greater height than the height component of the primary fuze point.

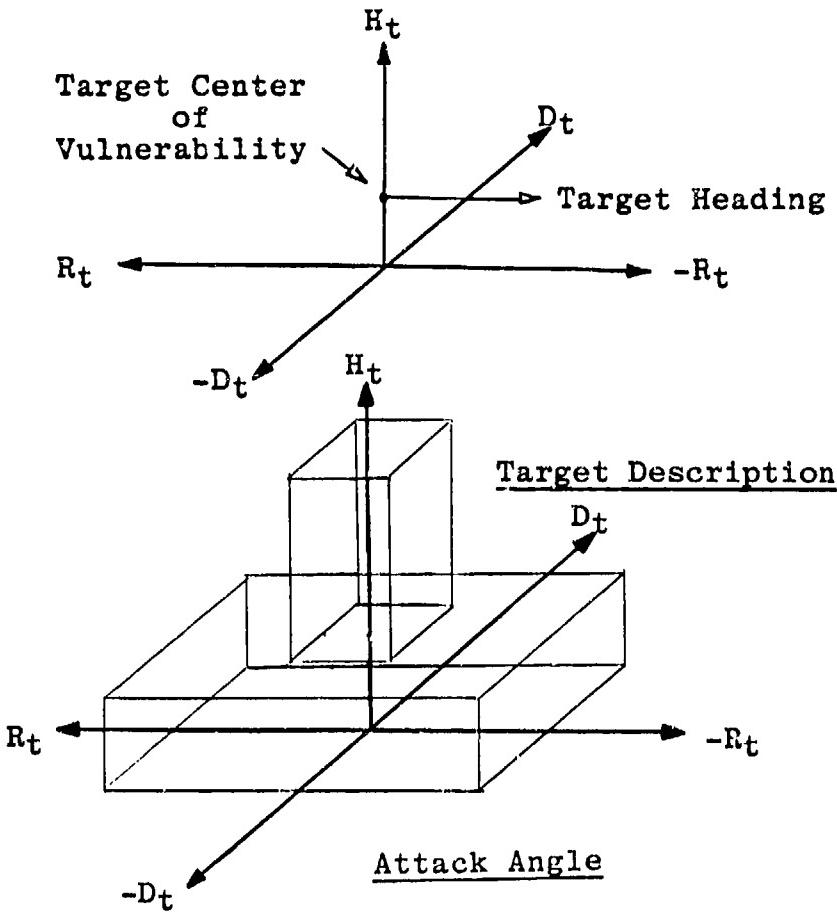
Target

The physical dimensions of the target are represented by a group (up to 5) of rectangular parallelepipeds (fig. 7) with the center of target vulnerability located over the origin of the ARP terminal coordinate system (R_t , D_t , H_t).

Burst Point

In all cases, once the fuzing point is found, a check is made to ascertain whether the target has been penetrated in order to reach that fuze point. If such penetration is found, the first penetration point is taken as the warhead functioning burst point (in this case, a direct hit burst point). Since the burst point is established in the rotated coordinate system (through the azimuth component of the attack angle), prior to determining kill effects, the burst point is rotated back into the target coordinate system.

Coordinate System



Attack Angle

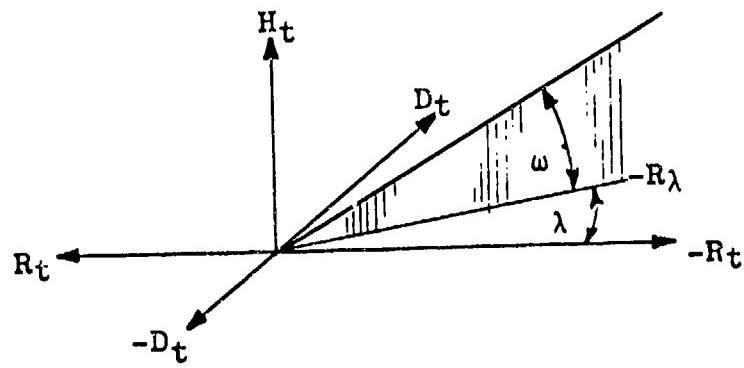


Figure 7. Target description

Direct Hit

If the burst point of the ARP is found to be at the surface of a parallelepiped representing a face of the target, a direct hit is deemed to have occurred.

Blast

Blast kills can be estimated for both the target vehicle and radar antenna.

Target Vehicle Blast

A table of blast radius versus burst height must be provided (fig. 8). If the burst point occurs within the radius specified for the determined height of burst, then a blast kill of the target vehicle is deemed to have occurred for that sample simulation with probability, p (fig. 9 and User Guide, app B).

Radar Blast

A function of the form illustrated in figure 10 must be provided for this option. This function defines radar blast kill probability as a function of range from the antenna to the burst point. For each simulation, radar blast kill is determined from the specified function.

Fragmentation

Fragmentation effects are determined from the results of preliminary MAE analysis of the fragmenting warhead. The MAE computer code is described in references 1 and 2. The MAE program computes conditional kill probabilities as a function of burst point proximity to target center, burst height, attack elevation angle, and projectile terminal velocity. With the MAE code for a given terminal scenario for each of several burst heights, a suitable representation of the fragmentation P_k function can be described. For each burst height, a P_k grid is computed which provides the basis for the construction of a P_k box grid about the target center. It is then a simple matter of interpolating in the range, deflection and height directions as well as for elevation angle to estimate the fragmentation P_k for the actual burst point (fig. 11). Fall-off P_k along the edges of the P_k box is assumed to be linear out to a specified limit; that is, a limit is specified in the range, deflection, and height directions at which the fragmentation P_k drops to zero.

It is important to note that the fragmentation kill probabilities generated by the MAE program are based on vulnerability data averaged over all attack azimuths. Also, P_k 's are determined by the MAE code by computation of the proximity

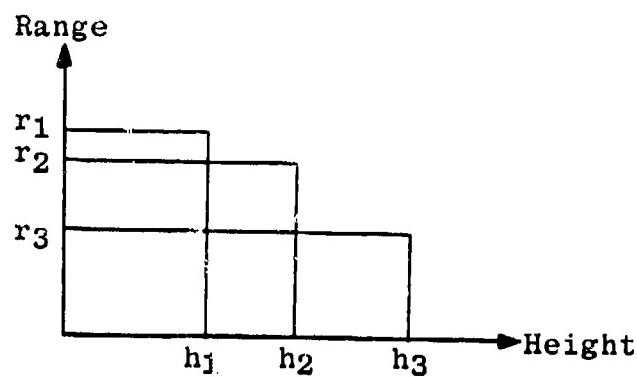


Figure 8. Blast radii vs height

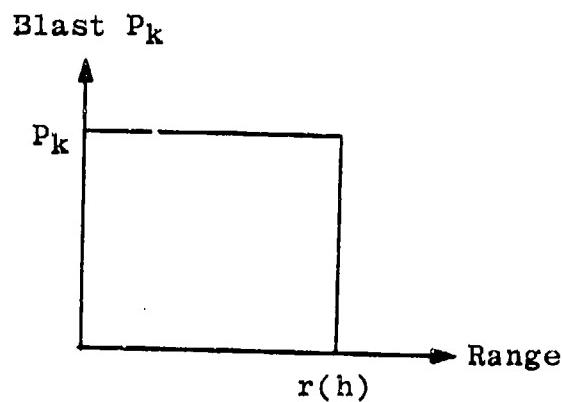


Figure 9. Blast kill probability vs height

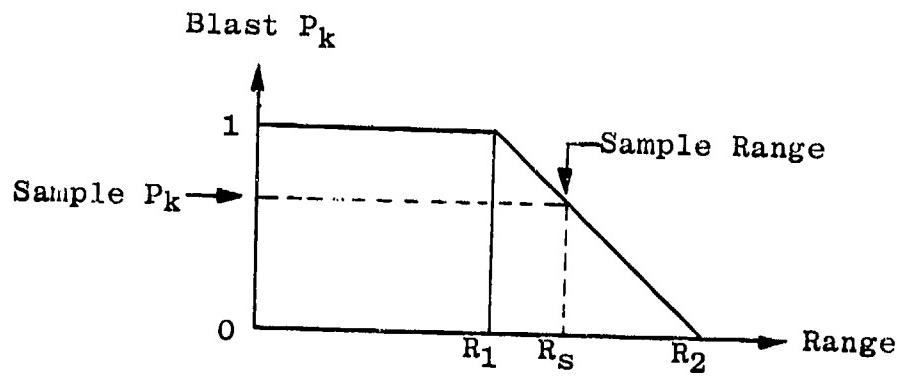


Figure 10. Radar blast function

Estimation of Fragmentation P_k

By Triple Interpolation

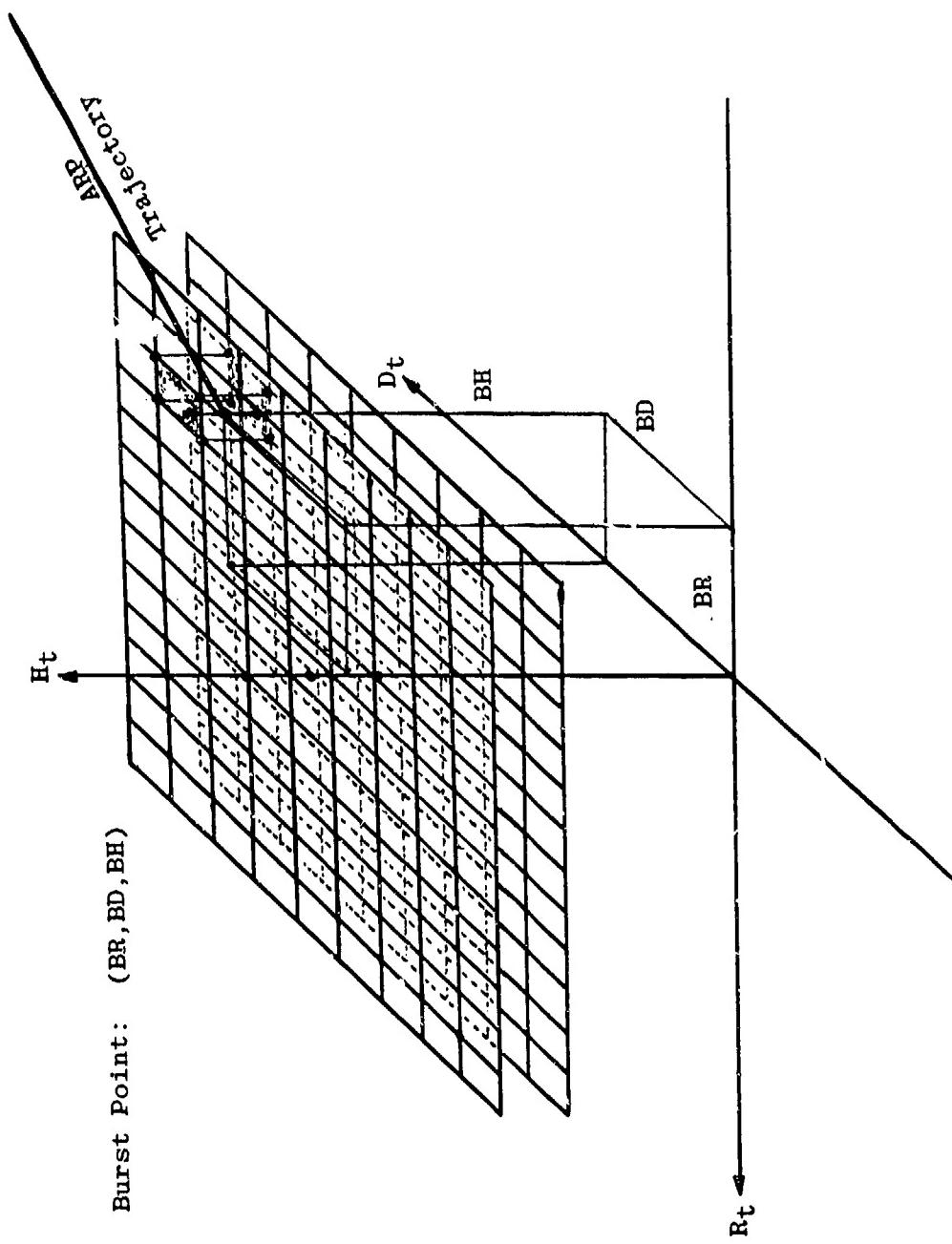


Figure 11. Fragmentation grid interpolation

of the burst point to the target center of vulnerability point. This point-to-point relationship is deficient for narrow spray angle munitions in close proximity to the target. Also, since ARPSIM assumes a particular attack azimuth, the assumption is made that, for the purposes of the ARPSIM model, the average vulnerability of the target can be used to represent the vulnerability for any particular attack azimuth.

As an alternative to the P_k grid box, the MAE program can be used to generate a P_k -versus-range function, where the P_k is averaged over all target azimuths. ARPSIM can utilize this function to interpolate for P_k based upon the range from the burst point to the target center. The P_k -versus-range function can be generated for various burst height and elevation angle combinations. This approach is not recommended with directional warheads.

When the MAE program is used, the blast option available with the MAE code should not be used.

MONTE CARLO ESTIMATES

The program flow procedures are followed for each simulation to provide estimates for direct hit, body blast, radar blast, and fragmentation effects in the form of kill probabilities, P_k . Estimates of these kill probabilities are computed by using

$$P_k(n) = \frac{\sum_{i=1}^n P_k(i)}{n}, \quad n = \text{sample size}$$

for each of the kill mechanisms. The combined kill probability is computed for each sample using:

$$P_k(i) = 1 - [1 - P_{DH}(i)] [1 - P_{RDR}(i)] [1 - P_{BLST}(i)] [1 - P_F(i)]$$

These overall kill probabilities are averaged for each individual component kill probability to give Monte Carlo estimates of the effectiveness of the individual kill mechanisms as well as the overall probability of defeating the target.

CONCLUSIONS

The ARPSIM model can be used to provide both weapon designers and effectiveness analysts with an assessment of the potential for the ARP system. As a design tool, ARPSIM provides insight into the contributions of guidance and fusing policies to the overall performance of the ARP warhead. ARPSIM does not simulate the guidance and control or radiation sensing mechanisms. ARPSIM does provide a means to parametrically assess the relative importance of various performance levels of the guidance, fuzing, and warhead functions. By providing

effectiveness information for a host of performance capabilities, ARPSIM is a useful tool to aid in exploiting those elements of the system which provide the greatest payoff in terms of system effectiveness. ARPSIM can also be utilized to provide data for systems analyses once performance criteria for guidance, fusing, and warhead functioning have been firmly established by weapon design.

The following specific assumptions and limitations are imbedded within the ARPSIM model:

1. Target is engaged in open fiat terrain.
2. ARP terminal trajectory is linear with the longitudinal axis of the projectile collinear with the trajectory.
3. The target configuration can be adequately represented by an aggregate of rectangular parallelepipeds.
4. Fragmentation effects can be estimated with the use of either a P_k box or a P_k -versus-range function generated by the material lethal area program based upon vulnerability data averaged over all azimuths.

RECOMMENDATIONS

The computer code follows a sequence of steps for each sample simulation. Any of these steps can be treated as a separate functional module (fig. 1). The degree of simulation detail can be changed by developing more complex modules to either increase simulation accuracy or expand modular function. The consequences of either improving the model's resolution or expanding its scope are an increase in computer processing time and a resultant increase in the cost of analysis. These consequences must be weighed against the advantages to be gained from the refinement of the model.

Some refinements which might be of merit include the direct computation of fragmentation effects (rather than use the results of precomputations with the MAE code) and the capability to define a complex target array consisting of a multiplicity of target elements.

REFERENCES

1. R. D. Webster, "An Overlay Computer Program for Fragmentation Reduction, Lethal Area, and Target Effects Computations," Information Report E2, Systems Effectiveness Branch, LCWSL, ARRAUDCOM, Dover, NJ, revised February 1980 by William Matzkowitz.
2. "Computer Program for General Full Spray Materiel MAE Computations, Vol 1, Users Manual," Manual 61 JTCG/ME-79-1-1, Joint Technical Coordinating Group for Munitions Effectiveness, 18 January 1979.

APPENDIX A

USER GUIDE

This user guide is intended to aid those who have access to the ARRADCOM CDC 6500/6600 central computing facility via INTERCOM and BATCH mode processing. Others who may wish to use or modify ARPSIM for operation on a different computer system should also find this guide informative and helpful.

For assessment of fragmentation effects with ARPSIM, it is first necessary to generate files containing fragmentation P_k data as determined by the materiel MAE code (ref 1). There are two alternate forms that the MAE-produced P_k data may take for use by ARPSIM:

1. A P_k grid where grids are defined for the ARP terminal elevation attack angle for up to four different burst heights.
2. A P_k versus range table defined for these same terminal conditions.

For directional fragmentation patterns, the P_k grid format provides a better estimate of the effects produced by the non-symmetry of the warhead effects pattern. P_k functions produced by the MAE code are developed as follows:

P_k Grid Function

Several options exist with the MAE code described in reference 1 which allow the user to define the bounds of the P_k grid in a variety of ways. It is important to note that ARPSIM is limited to a grid size of no more than 20 cells in either range or deflection directions. It is quite possible that fragmentation effects for an ARP warhead might exist at ranges far in excess of the actual miss distance from the target being attacked. For this reason, the user is advised to analyze the guidance errors and fuzing scheme being considered in order to determine practical limits to the size of the P_k grid. Input data for the MAE code are often in units of feet, whereas the P_k grid boundaries which are output are metric. Also, ARPSIM can be used with any consistent set of units, although it is recommended that the metric system be used. It is advisable then to pre-determine the practical range for a P_k grid and then use an option with the MAE code to define the limits of the P_k grid.

When using the OVRLAY code described in reference 1 to make MAE calculations, the MTRX option should be called for but not actually used; that is, the MTRX input data should consist of a blank card. For the user who is not familiar with the OVRLAY system of computer codes, it is a system that was established to provide users with the capability to make single computer runs beginning with raw fragmentation data continuing through MAE computations and the development of P_k grids, and culminating with estimates of artillery system effectiveness against certain target arrays. The overlay technique is used to combine a number of computer codes devoted to these analyses. The MTRX option signals the MAE code to produce a P_k grid on a file named TAPE4 in formats which are compatible with both the MTRX and ARPSIM codes. For this reason, the user should call for the MTRX option when using the OVRLAY code, and then provide only a blank card as the input for the MTRX code. By doing this, the user will normally terminate the OVRLAY code and will have defined a TAPE4 file consisting of a string of P_k grids, one for each burst height. It is advisable to save the TAPE4 file as a permanent file for future recall of the data, as necessary, when using the ARPSIM code.

If P_k grids are being generated for several (up to three) different attack elevations for use by ARPSIM, each elevation angle data set should be generated by a separate MAE run. Then, when recalling the P_k grid files, define the data on file TAPE2 for the lowest angle data, TAPE3 for the next lowest, and TAPE4 for the highest. Burst heights should always be computed in the order of lowest to highest.

For users who do not have access to the MAE code or who will use an alternate code to generate P_k grids, the files TAPE2, TAPE3, and TAPE4 should contain, sequentially, a card image data record in format (2I3) indicating the number of grid coordinates in range and deflection.

Next, are two card sets in format (10F7.1) where the first set defines the range coordinates of the grid boundaries and the second set defines the deflection boundaries. Boundaries are defined from lowest to highest values. Following these data sets are the P_k 's associated with the grid in format (10F7.5) where P_k 's are given first for the first range cell (lowest grid bracket) for each of the deflection cells (again, beginning with the lowest bracket) and proceeding through all range brackets in the same manner. All P_k grids are defined this way for each burst height in order of lowest to highest burst height.

P_k Versus Range

An average P_k versus range function (table) can be used if the number of ranges is no greater than 200. Format for data entry is (F8.3, F8.5) where the first item is range (usually in meters) followed by the corresponding average P_k . The MAE code can generate this table on a file named TAPE15. These files can be saved, like the grid files, and recalled when using the ARPSIM. These files when used other than with the MAE code or when recalling MAE-generated files, are defined like the grid files, i.e., lowest angle data on TAPE2, next lowest on TAPE3, and highest on TAPE4. Each burst height (up to four) has its own table defined beginning with the lowest burst height and stored sequentially on each file.

Following definition of the P_k functions on TAPE2, TAPE3, and TAPE4 (as required), the ARPSIM can be exercised using a teletype (TTY). Preliminary steps required to run ARPSIM on the ARRADCOM computer in INTERCOM mode are as follows:

INTERCOM Mode Setup

The following sequence is required to access the ARPSIM code and begin its execution:

```
LOGIN.  
...follow normal login procedures  
COMMAND - ETL,500.  
COMMAND - FETCH,ARP,BWEBSTER.  
COMMAND - ATTACH,T,TAPE1FILE,ID=your id.  
COMMAND - COPYBF,T,TAPE1.  
COMMAND - RETURN,T.
```

COMMAND - ATTACH,TAPE2,...
COMMAND - ATTACH,TAPE3,...
COMMAND - ATTACH,TAPE4,...
COMMAND - ARP

The sequence from ATTACH,T... through RETURN,T. is only required if a previously defined set of basic inputs is to be used as a basis for this run. Also, the sequence ATTACH,TAPE2,... through ATTACH,TAPE4,... is required only in accordance with the requirements to estimate fragmentation effects and the diversity of attack elevations required.

In response to the command ARP, the user will be given the opportunity to produce a summary input guide. Following that, the user will be asked whether a file named TAPE1 is to be used as the basis for input data. This option is provided as an aid to the user who expects to make several computer runs with the model using the same basic input data set. The ARPSIM code has a built-in input editing routine which continually redefines the file TAPE1 to be the current basic input data set. The user who wishes to make additional runs with a basic data set merely has to define the current data set and then, after ARPSIM has been run, the TAPE1 file is stored on a permanent file for later use as with the ATTACH,T... through RETURN,T. sequence described above. If a basic data set is being used, then the initial input conditions are listed. Then, in all cases, the user is asked to ENTER DATA OR END -. In response to this command the user begins to enter "word" type data to either initialize a data type or change a data type. Word type data which can be entered are defined according to general function in the section which follows. Formats are (A4,F10.4).

"Word" Type Data

This section is divided into functional areas as follows:

Guidance Data

- NGER,n. NGER signifies the number of guidance error data sets to be input. The value of n equals the number of different guidance error sets to be analyzed.
- NCEP,l. If guidance errors are input as standard deviations in both deflection and height, omit this set. If errors are input as CEP, then include this set. Note that in all cases errors are defined in a plane passing through a homing point and normal to the ARP flight path.

Fuzing Scheme

- FZAM,n. FZAM signifies the use of the fuzing angle primary fuze where n is the mean value of the fuze half-vertex angle; i.e., n is

the mean angle from the ARP trajectory to the fuzing glitter point at which fuzing will occur. Units are degrees.

- FZAS,n. FZAS signifies the standard deviation of fuzing angle associated with the mean value defined by FZAM, where the value n is the standard deviation. Units are degrees.
- FZTM,n. FZTM signifies the use of a linear (or time) fuze where the sign of the value of n indicates whether the fuze operates in the vertical direction or along the trajectory. A negative n signifies the vertical option. The value of n is the mean distance from the guidance plane (or initial fuzing point if used in conjunction with the FZAM option) in the negative range direction where fuzing occurs. With the vertical option, the distance is measured from the ground. A time fuze operating along the ARP trajectory can be simulated by converting the values to distances by using the known ARP terminal velocity.
- FZTS,n. FZTS defines the standard deviation associated with the FZTM data in all modes.
- PKPF,n. The value of n is the probability that the primary fuze (options described above) will function.
- PDVT,n. Selects the backup fuze option. The value for n is 0 for a PD (ground burst) backup and is the number of entries in a height versus probability table (up to 5 values) to define the VT fuze functioning distribution.
- GLTR,n. Specifies the glitter points used by the angular fuzing function option. If n is 0, the fuze functions relative to the point (0,0,TGTC) where TGTC is the center of target vulnerability. If n is non-zero, the fuze functions relative to one of the n input glitter points. A positive n signifies that the fuzing glitter point is selected randomly; a negative n signifies that the first glitter point encountered will cause fuzing.

Terminal Conditions

- OMEG,n. The elevation angle measured from the ground is chosen from a normal distribution with mean value n.
- OMGS,n. The standard deviation associated with OMEG is input as n.
- TGTC,n. The center of target vulnerability is input as a height above the origin at (0,0,n). If direct hit effects are not being analyzed (direct hit boxes are not defined), then the vehicle blast effects are determined based on the range from the burst point to (0,0,TGTC).

- DHAZ,n. The azimuth angle-of-attack is n and is measured from the negative range axis in the direction of the positive deflection axis. Units are degrees. To choose the azimuth uniformly random between 0 and 360 degrees, set n = -1.
- DUDR,n. The dud rate of ARP projectiles is given as n, where a 5% dud rate corresponds to n = 0.05.

General Conditions

- SAMP,n. The number of Monte Carlo samples is n.
- PRNT,l. Specifies that only a final summary of results is to be output.
- SRNG,n. Tables of average combined P_k can be output as a function of azimuth, elevation and range as well as averaged over non-zero results obtained in the angular bins for each range. The value for n is the upper limit (defaults to 100) for range information. The range scale is logarithmic and includes 10 bins, beginning with the minimum range obtainable (considering direct hit implications) and ending at n.

Fragmentation Effects

- PKNH,n. Specifies the number of heights, n, at which fragmentation effects are provided (either as P_k grids or P_k versus range tables). Must not exceed 4.
- PKNA,n. Specifies the number of elevations, n, for which fragmentation effects are provided. Must not exceed 3.
For n = 1, effects are on TAPE2.
For n = 2, effects are on TAPE2 for lowest angle data and on TAPE3 for highest angle data.
For n = 3, effects are on TAPE2 for lowest angle data, TAPE3 for middle angle data, and TAPE4 for highest angle data.
- FUNC,l. Selects option to use P_k versus range tables for fragmentation effects in place of the P_k grids.

Direct Hit Effects

- DHIT,n. Specifies the number of target boxes to be input to approximate the shape of the target for purposes of computing direct hit effects. Boxes are defined relative to (0,0,0) and the total number of boxes cannot exceed 5.

PKDH,n. Direct hit P_k if a direct hit is achieved. If $n = 0$, P_k is defaulted to one.

Blast Effects

PKBL,n. Specifies the blast P_k if the burst point is within a range specified by the BLST data of the surface of any direct hit box. If direct hit boxes are not used, then range is calculated to the point (0,0,TGTC).

BLST,n. Specifies the range from the direct hit surfaces or the point (0,0,TGTC) within which the blast P_k against the vehicle body is that given by the PKBL data. To enter a table of blast ranges versus burst height, enter a negative value for n which corresponds to the number of entries in the blast range versus height table (may not exceed 5).

RADR,l. Include to compute blast effects against radar antenna separately from vehicle blast.

End of Word Data

END Must always be included at the end of the "word"-type data entries.

After all "word"-type data have been entered, the code will ask for certain data which are required by some of the options chosen by the "word" cards. These additional input requirements are discussed in the following section. All data are free-formatted.

Guidance Data

Either pairs of deflection and height standard deviations are entered or, if NCEP,l. data is entered in the "word" section, then the guidance errors are input as CEP's.

The homing point coordinates follow the guidance error inputs. The homing point is generally the coordinates of the center of the radar antenna.

Direct Hit Boxes

The limits of the dimensions of each direct hit box are input for range, deflection, and height, respectively. For example, for a direct hit box centered at the origin and having a length of 20 meters, a width of 10 meters, and a height of 5 meters, this data would be input as -10,10,-5,5,0,5.

Radar Data

Radar antenna coordinates are entered for the purposes of radar blast P_k computation.

Following the entry of the radar coordinates, values are entered for two ranges, R1 and R2, which define the radar blast P_k function as being one out to R1 and declining linearly to zero at R2.

Fragmentation

Heights are entered beginning with the lowest value and corresponding to the burst heights used for the MAE computations. An additional height is input last and corresponds to that height at which all fragmentation P_k 's are zero.

Following the height data, two values are input corresponding to the distances beyond the edge of the P_k grids where the fragmentation P_k becomes zero in range and deflection, respectively.

Elevation angles are entered next, beginning with the lowest angle and corresponding to the angles for which the MAE code was run to produce the fragmentation P_k data.

VT Backup Fuzing

A table of probability of fuze functioning at height less than or equal to height, H, is used to generate VT fuzing data. Up to five heights are input followed by probabilities corresponding to the probability of fuze functioning between the respective height and the next lower height. Ideally, probability values should sum to unity.

Glitter Points

Glitter point coordinates are entered for each glitter point. All coordinates are relative to (0,0,0) of the target.

Blast Data (Vehicle)

If the blast-distance-versus-burst-height option is chosen (negative n on BLST,n data), then n pairs of blast distance, height are entered.

This concludes the input requirements for using the ARPSIM model. Word type data can be changed or input in any order. Required additional data will be

prompted from the user by the code. The user is always given the option of listing the current data set (with the exception of the fragmentation P_k data) or changing the data set prior to actual computations. When the computations are completed for all cases, the user is given the opportunity to run additional cases based on the current data sets.

APPENDIX B

EXAMPLE

3 /

27 / 28

The following example, provided as a supplement to the User Guide in Appendix A, denotes the type of material generated for a typical ARPSIM run:

* ANTI-RADIATION SIMULATION PROGRAM - 9/1/80 *

* NOTE: ALL COORDINATES ARE DEFINED RELATIVE TO *
* ORIGIN AT GROUND ZERO OF TARGET. *
* COORDINATE SYSTEM IS RECTANGULAR. *
* TARGET HEADING IS NEGATIVE RANGE. *
* DRIVER SIDE (L) IS POSITIVE DEFLECTION. *
* HEIGHT IS MEASURED FROM GROUND. *

DATE - 08/27/80
TIME - 13.47.13.

*DO YOU WANT A LISTING OF CODE NAMES? 'Y'

OMEG - MEAN ATTACK ANGLE
OMGS - ATTACK ANGLE STD DEV
* NOTE: FOLLOWING GUIDANCE ERROR PARAMETERS*
* (SIGD,SIGH) ARE MEASURED*
* IN PLANE NORMAL TO TRAJECTORY AND*
* PASSING THROUGH HOMING POINT*
NGER - NUMBER OF GUIDANCE ERRORS TO CONSIDER
* ENTER HOMING POINT (R,D,H), GUIDANCE*
* ERRORS ARE DISTRIBUTED ABOUT HOMING PT.*
NCEP - 1.. IF CEP IS INPUT FOR GUIDANCE ERROR SIGMAS
FUNC - 1.. IF OPTION TO USE PK VS. RANGE DAMAGE
* IN PLACE OF PK BOX FUNCTION IS SELECTED*
* YOU MUST DEFINE PK VS R DATA FOR EACH*
* HEIGHT LAYER SPECIFIED BY PKMH CARD*
* AND EACH ANGLE SPECIFIED BY*
* PKNA CARD.*
FZAM,FZAS,FZTM,FZTS - FUZING ERROR OPTIONS
* FZAM - MEAN ANGLE AT WHICH FUZING OCCURS ON*
* INTERCEPT*
* FZAS - STD DEV ASSOCIATED WITH FZAM*
NOTE: FOR UNIFORM FUZING ANGLE BETWEEN FZAM
* AND FZAS, ENTER A NEGATIVE VALUE FOR FZAM*
* FUZE ANGLE WILL BE CHOSEN UNIFORMLY RANDOM*
BETWEEN POSITIVE FZAM AND FZAS
NOTE: FUZING PLANE PASSES THROUGH FUZING GLITTER
* POINT NORMAL TO SAMPLE TRAJECTORY*
* FZTM - MEAN DISTANCE FROM GUIDANCE PLANE AT WHICH*
* FUZING WILL OCCUR ALONG TRAJECTORY*
*NOTE: ENTER A NEGATIVE FZTM FOR HEIGHT FUZING *
* WITH MEAN HEIGHT ABS(FZTM) *

* FZTS - STD DEU ASSOCIATED WITH FZTH
 * SAMP - SAMPLE SIZE
 * PKNH - NUMBER OF HEIGHTS AT WHICH FRAGMENTATION
 * PK DATA WILL BE DEFINED
 * NOTE: PKNH < 5
 * PKHA - NUMBER OF ELEVATION ANGLES FRAGMENTATION
 * PK DATA WILL BE DEFINED FOR
 * NOTE: PKHA < 4
 * PKPF - PROBABILITY OF PRIMARY FUZE FUNCTIONING
 * PDUT - 0, FOR PD BACKUP, NOT FOR UT BACKUP FUZE
 * WHERE NUT = NUMBER OF UT BURST HEIGHTS
 * GLTR - 0, IF PRIMARY FUZE FUNCTIONS RELATIVE TO
 * CENTER OF TARGET, MGLT IF PRIMARY FUZE
 * FUNCTIONS RELATIVE TO ANY ONE OF MGLT
 * EQUIALLY LIKELY GLITTER POINTS
 * SET MGLT NEGATIVE TO PICK FIRST
 * POINT ENCOUNTERED.
 * SRNG - MAXIMUM RANGE FOR COMPUTING PK US RANGE
 * PRNT - 1, TO PRINT SUMMARY ONLY, 0, OTHERWISE
 * DBUG - 0, TO PRINTOUT PROGRAM DEBUGGING DATA
 * DBUG - 1, GUIDANCE & FUZING DATA
 * DBUG - 2, DIRECT HIT PENETRATION DATA
 * DBUG - 3, HOMING ANGLE DATA
 * DBUG - 4, PK BOX DATA
 * DBUG - 5, PK GRIDS
 * DBUG - 6, PK US R DATA
 * TGTC - HEIGHT OF TARGET CENTER ABOVE GROUND
 * DUDR - DUD RATE OF PROJECTILE, EXPRESSED AS A FRACTION
 * DHIT - DIRECT HIT OPTION, NUMBER OF TARGET BOXES
 * IF DHIT IS OMITTED AND BLST IS INCLUDED,
 * BLST IS RADIUS FROM (0,0,TGTC) WITHIN
 * WHICH PKBLST = 1.
 * PKDH - DIRECT HIT PK (0. = 1.)
 * PKBL - BLAST PK (0. = 1.)
 * RADR - 1., DEFINE FUNC FOR BLAST KILL OF RADAR ONLY
 * AND READ IN RADAR ANTENNA COORDINATES.
 * TO DEFINE FUNC, SPECIFY R1 AND R2,
 * WHERE BLAST PK IS 1 OUT TO R1 AND
 * DECLINES LINEARLY TO 0 AT R2.
 * DHAZ - AZIMUTH ANGLE OF ATTACK OFF FRONT OF TARGET
 * TOWARD DRIVER SIDE. SET TO -1, FOR RANDOM
 * BLST - BLAST RADIUS WITHIN WHICH VEHICLE PK=PKBL
 * NOTE: TO ENTER BLAST RADII US, BLAST HEIGHT,
 * ENTER NEGATIVE NUMBER OF BLST, NOT PAIRS
 * IN PLACE OF VALUE OF BLST. PAIRS OF
 * BLAST,HGT ARE ENTERED IN ASCENDING ORDER
 * OF HEIGHT.
 * COORDINATE SYSTEM IS RECTANGULAR.
 * TARGET HEADING IS NEGATIVE RANGE.
 * DRIVER SIDE (LEFT) IS POSITIVE DEFLECTION.
 * HEIGHT IS MEASURED FROM GROUND.
 * ENTER DATA BY ENTERING CODE NAME
 * FOLLOWED BY A COMMA AND THE VALUE IN FLOATING
 * POINT FORMAT. TO END DATA ENTRY, ENTER
 * THE WORD END IN COLUMNS 1-3
 * DO YOU WISH TO INITIALIZE DATA FROM SAVED
 * DATA FILE (TAPE1)? 'Y'

*INITIAL INPUTS - *
 FZAM 70.000
 PKDH 1.000
 PKBL 1.000
 FZAS 10.000
 OMEG 10.000
 NGER 3.000
 NCEP 1.000
 FUNC 1.000
 DHIT 2.000
 SAMP 100.000
 PKNH 4.000
 PKHA 3.000
 PDUT 5.000
 PKPF .950
 GLTR 3.000
 SRNG 100.000
 TGTC 10.000
 DUDR .050
 BLST 3.000
 END

3. 6. 9.
 0. 0. 10.
 -5. 5. -5. 5. 0. 10. -2. 2. -2. 2. 10. 20.
 4. 8. 12. 16.
 0. 10. 20.
 2. 4. 6. 8. 10.
 .2. .2. .2. .2.
 -5. -5. 10. -5. 5. 0. -2. 2. 20.

*DO YOU WANT TO CHANGE ANY DATA? - 'Y

*ENTER DATA OR END - 'RBDR.1.

*ENTER DATA OR END - 'END

RADAR DATA -
ENTER RADAR ANTENNA COORDINATES (R,D,H) RELATIVE
TO TARGET GROUND ZERO. -0.,0.,20.

ENTER R1,R2, WHERE RADAR BLAST PK=1
OUT TO R1 AND DECLINES LINEARLY
TO ZERO AT R2 -10.,20.

*DO YOU WANT CURRENT DATA LISTED? 'N

*DO YOU WANT TO CHANGE ANY DATA? - 'N

XX

FINAL RESULTS

PK = .7810 PKED = .0350 NSAMP = 100

XX

*DO YOU WANT PK VS R, ALPHA, BETA? 'Y

PK	R	ALPHA	BETA
1.0000	11.8	60.0 - 90.0	60.0 - 75.0
1.0000	11.8	120.0 - 150.0	15.0 - 30.0
1.0000	11.8	120.0 - 150.0	30.0 - 45.0
1.0000	11.8	120.0 - 150.0	45.0 - 60.0
1.0000	11.8	150.0 - 182.0	45.0 - 60.0
1.0000	11.8	150.0 - 182.0	60.0 - 75.0
1.0020	11.8	180.0 - 210.0	60.0 - 75.0
1.0000	11.8	210.0 - 240.0	30.0 - 45.0
1.0020	11.8	210.0 - 240.0	45.0 - 60.0
.7433	12.7	120.0 - 150.0	30.0 - 45.0
1.0000	12.7	120.0 - 150.0	45.0 - 60.0
.9456	12.7	150.0 - 180.0	45.0 - 60.0
1.0000	12.7	150.0 - 180.0	60.0 - 75.0
1.0000	12.7	210.0 - 240.0	45.0 - 60.0
1.0000	12.7	210.0 - 240.0	60.0 - 75.0
1.0000	12.7	210.0 - 240.0	75.0 - 90.0
.9851	14.1	120.0 - 150.0	45.0 - 60.0
1.0000	14.1	120.0 - 150.0	60.0 - 75.0
1.0000	14.1	150.0 - 180.0	45.0 - 60.0
1.0000	14.1	150.0 - 180.0	60.0 - 75.0
1.0000	14.1	210.0 - 240.0	45.0 - 60.0
.9814	16.2	120.0 - 150.0	45.0 - 60.0
.7858	16.2	150.0 - 180.0	30.0 - 45.0
.8309	16.2	150.0 - 180.0	45.0 - 60.0
1.0000	16.2	150.0 - 180.0	60.0 - 75.0
.6365	16.2	160.0 - 210.0	30.0 - 45.0
1.0000	16.2	210.0 - 240.0	45.0 - 60.0
1.0000	16.2	210.0 - 240.0	60.0 - 75.0
.8806	19.7	120.0 - 150.0	45.0 - 60.0
1.0000	19.7	120.0 - 150.0	60.0 - 75.0
.5643	19.7	150.0 - 180.0	30.0 - 45.0
.7680	19.7	150.0 - 180.0	45.0 - 60.0
.4257	19.7	180.0 - 210.0	30.0 - 45.0
.0828	19.7	210.0 - 240.0	45.0 - 60.0
1.0000	19.7	210.0 - 240.0	60.0 - 75.0
.2159	25.1	150.0 - 180.0	30.0 - 45.0
.1064	33.5	150.0 - 180.0	15.0 - 30.0
.1227	33.5	180.0 - 210.0	15.0 - 30.0
.0015	46.7	0.0 - 10.0	0.0 - 15.0

AUG 19 US. R

1.0000	11.1
.9676	12.1
.9000	14.1
.9036	16.1
.7302	19.1
.2159	25.1
.1184	33.1
.0315	46.1

ANSWER The answer is 1000.

'FINAL RESULTS'

PK = .6970 PKSD = .0409 NSAMP = 100

'DO YOU WANT PK US R, ALPHA, BETA? 'V

PK	R	ALPHA	BETA*
1.0000	11.8	120.0 - 150.0	15.0 - 30.0
1.0020	11.8	120.0 - 150.0	30.0 - 45.0
1.0030	11.8	120.0 - 150.0	45.0 - 60.0
1.0040	11.8	220.0 - 150.0	60.0 - 75.0
1.0050	11.8	150.0 - 180.0	30.0 - 45.0
1.0060	11.8	150.0 - 180.0	45.0 - 60.0
1.0070	11.8	180.0 - 240.0	30.0 - 45.0
1.0080	12.7	120.0 - 150.0	30.0 - 45.0
.8573	12.7	150.0 - 180.0	15.0 - 30.0
1.0090	12.7	150.0 - 180.0	60.0 - 75.0
1.0000	12.7	210.0 - 240.0	45.0 - 60.0
1.0000	14.1	90.0 - 120.0	60.0 - 75.0
1.0000	14.1	120.0 - 150.0	45.0 - 60.0
1.0000	14.1	120.0 - 150.0	60.0 - 75.0
1.0077	14.1	150.0 - 180.0	45.0 - 60.0
1.0000	14.1	150.0 - 180.0	60.0 - 75.0
1.0000	14.1	210.0 - 240.0	45.0 - 60.0
1.0000	16.2	90.0 - 120.0	60.0 - 75.0
.9349	16.2	120.0 - 150.0	45.0 - 60.0
.6972	16.2	150.0 - 180.0	30.0 - 45.0
.8547	16.2	150.0 - 180.0	45.0 - 60.0
1.0000	16.2	150.0 - 180.0	60.0 - 75.0
.4121	16.2	180.0 - 210.0	15.0 - 30.0
1.0000	16.2	210.0 - 240.0	45.0 - 60.0
1.0001	16.2	210.0 - 240.0	60.0 - 75.0
.9039	19.7	120.0 - 150.0	45.0 - 60.0
1.0000	19.7	120.0 - 150.0	60.0 - 75.0
1.0000	19.7	120.0 - 150.0	75.0 - 90.0
.3305	19.7	150.0 - 180.0	15.0 - 30.0
.3953	19.7	180.0 - 210.0	30.0 - 45.0
.8526	19.7	150.0 - 180.0	45.0 - 60.0
1.0000	19.7	150.0 - 180.0	60.0 - 75.0
1.0000	19.7	150.0 - 180.0	75.0 - 90.0
.2522	19.7	180.0 - 210.0	0.0 - 15.0
.9565	19.7	210.0 - 240.0	45.0 - 60.0
1.0000	19.7	210.0 - 240.0	60.0 - 75.0
1.0000	25.1	120.0 - 150.0	60.0 - 75.0
1.0000	25.1	120.0 - 150.0	75.0 - 90.0
.2450	25.1	150.0 - 180.0	15.0 - 30.0
.9414	25.1	180.0 - 210.0	45.0 - 60.0
.2338	25.1	180.0 - 210.0	60.0 - 75.0
.1777	25.1	180.0 - 210.0	15.0 - 30.0
.2979	25.1	180.0 - 210.0	30.0 - 45.0
1.0000	25.1	210.0 - 240.0	60.0 - 75.0
.1079	33.5	150.0 - 180.0	15.0 - 30.0
.0794	33.5	150.0 - 180.0	45.0 - 60.0
.1056	33.5	180.0 - 210.0	2.0 - 15.0
.1447	33.5	180.0 - 210.0	15.0 - 30.0
.9911	33.5	210.0 - 240.0	30.0 - 45.0
.2361	46.7	0.0 - 30.0	0.0 - 15.0
.0299	46.7	150.0 - 210.0	0.0 - 15.0
.0232	46.7	180.0 - 210.0	0.0 - 15.0
.0073	67.5	0.0 - 30.0	0.0 - 15.0
.0176	67.5	150.0 - 180.0	0.0 - 15.0
.0149	67.5	180.0 - 210.0	0.0 - 15.0
.0015	100.0	150.0 - 180.0	0.0 - 15.0

XX

'AUG PK US. R'

1.3200	11.8
.9420	12.7
.5797	14.1
.5215	16.2
.8305	19.7
.6480	25.1
.2942	33.5
.0365	46.7
.0132	67.5
.0015	100.0

XX

'FINAL RESULTS'

PK = .6983 PKSD = .0397 NSAMP = 100

XX

'DO YOU WANT PK US R, ALPHA, BETA? 'Y'

PK	R	ALPHA	BETA
1.0000	11.8	120.0 - 150.0	15.0 - 30.0
1.0000	11.8	120.0 - 150.0	30.0 - 45.0
1.0000	11.8	120.0 - 150.0	45.0 - 60.0
1.0000	11.8	150.0 - 180.0	45.0 - 60.0
.6204	11.8	210.0 - 240.0	15.0 - 30.0
1.0000	11.8	210.0 - 240.0	30.0 - 45.0
1.0000	12.7	120.0 - 150.0	30.0 - 45.0
1.0000	12.7	120.0 - 150.0	45.0 - 60.0
.5632	12.7	150.0 - 180.0	0.0 - 15.0
.7695	12.7	150.0 - 180.0	30.0 - 45.0
.8211	12.7	150.0 - 180.0	45.0 - 60.0
1.0000	12.7	210.0 - 240.0	45.0 - 60.0
.4675	14.1	30.0 - 60.0	15.0 - 30.0
1.0000	14.1	120.0 - 150.0	45.0 - 60.0
.6169	14.1	150.0 - 180.0	15.0 - 30.0
1.0000	14.1	210.0 - 240.0	45.0 - 60.0
.9818	16.2	120.0 - 150.0	45.0 - 60.0
1.0000	16.2	120.0 - 150.0	60.0 - 75.0
.4140	16.2	150.0 - 180.0	15.0 - 30.0
.6730	16.2	150.0 - 180.0	30.0 - 45.0
.8445	16.2	150.0 - 180.0	45.0 - 60.0
1.0000	16.2	150.0 - 180.0	60.0 - 75.0
1.0000	16.2	210.0 - 240.0	45.0 - 60.0
1.0000	16.2	210.0 - 240.0	60.0 - 75.0
.9435	19.7	120.0 - 150.0	45.0 - 60.0
1.0000	19.7	120.0 - 150.0	60.0 - 75.0
1.0000	19.7	120.0 - 150.0	75.0 - 90.0
.4614	19.7	150.0 - 180.0	15.0 - 30.0
.3359	19.7	150.0 - 180.0	30.0 - 45.0
.9249	19.7	150.0 - 180.0	45.0 - 60.0
1.0000	19.7	150.0 - 180.0	60.0 - 75.0
1.0000	19.7	150.0 - 180.0	75.0 - 90.0
1.0000	19.7	210.0 - 240.0	60.0 - 75.0
1.0000	25.1	120.0 - 150.0	60.0 - 75.0
1.0000	25.1	120.0 - 150.0	75.0 - 90.0
.2500	25.1	150.0 - 180.0	15.0 - 30.0
.0988	25.1	150.0 - 180.0	60.0 - 75.0
.2247	25.1	180.0 - 210.0	0.0 - 15.0
.1878	25.1	180.0 - 210.0	15.0 - 30.0
.9106	25.1	180.0 - 210.0	60.0 - 75.0
1.0000	25.1	210.0 - 240.0	60.0 - 75.0
.9755	33.5	120.0 - 150.0	60.0 - 75.0
.1173	33.5	150.0 - 180.0	0.0 - 15.0
.1316	33.5	150.0 - 180.0	15.0 - 30.0
.2503	33.5	150.0 - 180.0	45.0 - 60.0
.5454	33.5	150.0 - 180.0	60.0 - 75.0
.5349	33.5	180.0 - 210.0	60.0 - 75.0
1.0000	33.5	210.0 - 240.0	60.0 - 75.0
.0498	46.7	150.0 - 180.0	0.0 - 15.0
.0450	46.7	180.0 - 210.0	0.0 - 15.0
.0389	67.5	150.0 - 180.0	0.0 - 15.0
.0329	67.5	180.0 - 210.0	0.0 - 15.0
.0009	100.0	150.0 - 180.0	0.0 - 15.0

*AUG PK VS. %:

.9578	11.8
.9193	12.7
.9169	14.1
.9130	16.2
.9061	19.7
.8510	25.1
.4844	33.5
.0482	46.7
.0059	67.5
.0009	100.0

RESULTS FOR FOLLOWING CONDITIONS -

ITEM	MEAN	STD DEV
------	------	---------

ELEVATION	10.0000	0.0000
FUZE ANGLE	70.0000	10.0000
LINEAR FUZE	0.0002	0.0000
AZIMUTH	0.0002	0.0000
SAMPLE SIZE -	100	

HOMING POINT COORDINATES (R,D,H) = 0.0, 0.0, 10.0

ERROR DATA	PK	PKFRAG	PKRADR	PKDHIT	PKBLST
CEP - 3.0	.7816	.4173	.0003	.0700	.5400
CEP - 6.0	.6579	.3299	.5617	.0500	.4000
CEP - 9.0	.6983	.2613	.5725	.0100	.3200

*DO YOU WISH TO RUN ANOTHER CASE? *N

A description of the material produced by this particular ARPSIM run follows:

Header information is printed, including the time and date of the run. The user is asked whether a listing of input code names is desired (as an aid to generating a proper set of inputs). In this example, the code names are printed. Next, the user is given the option of starting with a previously developed set of inputs which can be changed by a built-in input editing routine. That option is invoked for this example. Note that a file named TAPE1 must be defined which contains this data prior to running ARPSIM. A listing of initial data conditions is provided next. The user is then asked whether any data changes are required.

In this example the user desires to add the capability to estimate radar blast effects. Note that only changed data need be entered at this point. The code then asks for additional information required by the added data. Having fulfilled the data requirements, the user is given the option of listing the entire data set again. Following this, the user is given the option of making any additional changes or corrections to the data set. In this example no additional changes are requested.

Before proceeding with the discussion of the ARPSIM results for this case, a brief run-through is given of the input data set. The FZAM data specifies a fuze angle option with a mean value of 70 degrees for the fuze angle. The FZAS code specifies a 10-degree standard deviation for the fuze angle from simulation to simulation. The PKDH and PKBL data indicate direct hit and vehicle blast P_k 's, respectively. Attack elevation of 10 degrees is specified by the OMEG card. NGER indicates three different sets of guidance errors will be analyzed, and NCEP indicates that guidance errors will be input as CEP. FUNC specifies that the fragmentation P_k 's will be estimated from interpolations in a set of P_k versus range tables generated by the MAE code for a combination of burst height and elevation angles.

Up to three elevation angle sets can be provided on files TAPE2, TAPE3, and TAPE4. If only a single elevation angle data set is provided, then only TAPE2 is required. Two elevation angles require both TAPE2 and TAPE3. Each file contains P_k versus range for identical burst heights, beginning with the lowest burst height. That is, if four burst heights have been analyzed by the MAE code (the maximum allowable by ARPSIM), each file will contain four P_k versus range tables, one for each burst height beginning with the lowest height and progressing to the highest.

In this example, four burst heights were considered for each of three angles of fall (elevation angles) as specified by the PKNH and PKNA codes, respectively. SAMP provides the number of simulations to run for each case. PDVT specifies that a VT backup fuze is being considered where the height of burst distribution for the backup fuze will be typified at five burst heights. PKPF specifies that the probability that the primary fuze functions is 0.95. GLTR specifies that three glitter points for primary fuzing exist. SRNG gives the maximum range for a P_k versus range table to be generated based upon the results of the ARPSIM run. TGTC provides that the center of target vulnerability is located at 10 (in this case meters) above the target origin (0,0,0). DUDR specifies a projectile dud rate of 5%. BLST provides a blast radius from the TGTC point within which the P_k for vehicle blast effect is as stated on the PKBL data above.

The END code signifies the end of the word type data. The numbers 3., 6., and 9. specify the guidance error CEP's. Following this are the homing point coordinates (0,0,10), and the limits in range, deflection and height of the two direct hit target description boxes. Burst heights and angles of fall (elevations) utilized by the MAE code in generating the P_k versus range tables are specified next. Then the heights and probabilities associated with the backup fuzing function are listed. Finally, glitter point coordinates are specified.

Final results are given as the combined kill probability, the standard deviation of kill probability and the sample size upon which these numbers are based. The user is given the option of listing the generated hemispheric distribution of computed combined P_k 's, where the angle alpha denotes azimuth and beta denotes elevation from the burst point to (0,0,0). The range specified is also the range from the burst point to the origin (0,0,0). These hemispheric data (only the positive elevation angles are considered since negative angles would imply a burst below ground) are averaged over all angular bins for which burst points were analyzed to provide a table of average P_k versus range.

The final results are repeated for each case and followed by a summary of the results for each type P_k considered together with the corresponding error data for that case.

After all results have been given for all cases specified, the user is given the opportunity to run additional cases, based upon the same data set. In all cases, the contents of the file TAPE1 are always the last data set considered. Consequently, if the user wishes to make additional runs with ARPSIM at a later time using the same basic data set, then after the current runs with ARPSIM are finished, the file TAPE1 can be saved as a starting point for future runs.

TAPE1 can be retained as a permanent file. However, for access at a later date, this TAPE1 must be attached with a different local file name. Then this local file name is copied to a new file named TAPE1. These steps are necessary because the ARPSIM code changes the contents of the file TAPE1.

APPENDIX C
FORTRAN LISTING

Note: The following FORTRAN listing is subject to changes as dictated by improvements or modifications to the ARPSIM model.

PROGRAM ARP 73/74 OPT=1

FTN 4.8+508 03/13/81 08.28.23 PAGE

03/13/81 08.23.23

WRITE (6,*) * NOTE: FOLLOWING GUIDANCE ERROR PARAMETERS" 00067C
 WRITE (6,*) * (SIGN SIGH) ARE MEASURED"
 WRITE (6,*) * IN PLANE NORMAL TO TRAJECTORY AND" 000680
 WRITE (6,*) * PASSING THROUGH HOMING POINT" 00069C
 WRITE (6,*) * NGER - NUMBER OF GUIDANCE ERRORS TO CONSIDER" 000700
 WRITE (6,*) * ENTER HOMING POINT (R,D,H). GUIDANCE" 000710
 WRITE (6,*) * ERRORS ARE DISTRIBUTED ABOUT HOMING PT." 000720
 WRITE (6,*) * NCEP = 1.. IF CEP IS INPUT FOR GUIDANCE ERROR SIGMAS" 000730
 WRITE (6,*) * FZAM,FZAS,FZTM,FZTS - FUZING ERROR OPTIONS" 000740
 WRITE (6,*) * FZAM - MEAN ANGLE AT WHICH FUZING OCCURS ON" 000750
 WRITE (6,*) * INTERCEPT" 000760
 WRITE (6,*) * FZAS - STD DIV ASSOCIATED WITH FZAM" 000770
 WRITE (6,*) * "NOTE: FUZE ANGLE IS CONSTRAINED TO (0,PI)" 000780
 WRITE (6,*) * "NOTE: FOR UNIFORM FUZING ANGLE BETWEEN FZAM" 000790
 WRITE (6,*) * AND FZAS, ENTER A NEGATIVE VALUE FOR FZAM" 000800
 WRITE (6,*) * FUZE ANGLE WILL BE CHOSEN UNIFORMLY RANDOM" 000810
 WRITE (6,*) * BETWEEN POSITIVE FZAM AND FZAS" 000820
 WRITE (6,*) * FOR TIME-TO-GO FUZE, ENTER NEGATIVE FZAS" 000830
 WRITE (6,*) * "NOTE: FUZING PLANE PASSES THROUGH FUZING GLITTER" 000840
 WRITE (6,*) * POINT NORMAL TO SAMPLE TRAJECTORY" 000850
 WRITE (6,*) * FZTM - MEAN DISTANCE FROM GUIDANCE PLANE AT WHICH" 000860
 WRITE (6,*) * FUZING WILL OCCUR ALONG TRAJECTORY" 000870
 WRITE (6,*) * "NOTE: ENTER A NEGATIVE FZTM FOR HEIGHT FUZING" 000880
 WRITE (6,*) * WITH MEAN HEIGHT ABS(FZTM)" 000890
 WRITE (6,*) * FZTS - STD DEV ASSOCIATED WITH FZTM" 000900
 WRITE (6,*) * SAMP - NUMBER OF HEIGHTS AT WHICH FUZING OCCURS" 000910
 WRITE (6,*) * PKNH - NUMBER OF HEIGHTS AT WHICH FRAGMENTATION" 000920
 WRITE (6,*) * PK DATA WILL BE DEFINED" 000930
 WRITE (6,*) * NOTE: PKNH < 9" 000940
 WRITE (6,*) * PKPF - PROBABILITY OF PRIMARY FUZE FUNCTIONING" 000950
 WRITE (6,*) * FDVT - O. FOR PC BACKUP, NVT FOR VT BACKUP FUZE" 000960
 WRITE (6,*) * WHERE NVT = NUMBER OF VT BURST HEIGHTS" 000970
 WRITE (6,*) * GLTR - O. IF PRIMARY FUZE FUNCTIONS RELATIVE TO" 000980
 WRITE (6,*) * CENTER OF TARGET, NGLT IF PRIMARY FUZE" 000990
 WRITE (6,*) * FUNCTIONS RELATIVE TO ANY ONE OF NGLT" 001000
 WRITE (6,*) * EQUALLY LIKELY GLITTER POINTS." 001010
 WRITE (6,*) * SET NGLT NEGATIVE TO PICK FIRST" 001020
 WRITE (6,*) * POINT ENCOUNTERED." 001030
 WRITE (6,*) * SRNG - MAXIMUM RANGE FOR COMPUTING PK VS RANGE" 001040
 WRITE (6,*) * PRNT - 1. TO PRINT SUMMARY ONLY. 0. OTHERWISE" 001050
 WRITE (6,*) * DBUG - 6. TO PRINTOUT PROGRAM DEBUGGING DATA" 001060
 WRITE (6,*) * DBUG = 1, GUIDANCE & FUZING DATA" 001070
 WRITE (6,*) * DBUG = 2, DIRECT HIT PENETRATION DATA" 001080
 WRITE (6,*) * DBUG = 4, PK BOX DATA" 001090
 WRITE (6,*) * DBUG = 5, PK GRIDS" 001100
 WRITE (6,*) * DBUG = 6, PK VS R DATA" 001110
 WRITE (6,*) * IGTC - HEIGHT OF TARGET CENTER ABOVE GROUND" 001120
 WRITE (6,*) * DUDR - DUD RATE OF PROJECTILE, EXPRESSED AS A FRACTIO 001130
 CN" 001140
 WRITE (6,*) * DHIT - DIRECT HIT OPTION, NUMBER OF TARGET BOXES" 001150
 WRITE (6,*) * IF DHIT IS OMITTED AND BLST IS INCLUDED," 001160
 WRITE (6,*) * BLST IS RADIUS FROM (0,0,TGTC) WITHIN" 001170
 WRITE (6,*) * WHICH PKBLST = 1." 001180
 WRITE (6,*) * PKCH - DIRECT HIT PK (0. = 1.)" 001190
 WRITE (6,*) * PKEL - BLAST PK (0. = 1.)" 001200
 WRITE (6,*) * RADR - 1. DEFINE FUNC FOR BLAST KILL OF RADAR ONLY" 001210
 WRITE (6,*) * AND READ IN RADAR ANTENNA COORDINATES." 001220
 WRITE (6,*) * 001230

PROGRAM	ARP	73/74	OPT=1	FTN 4.8+508	03/13/81	08.28.23	PAGE
1115				TO DEFINE FUNC, SPECIFY R1 AND R2, WHERE BLAST PK IS 1 OUT TO R1 AND- DECLINES LINEARLY TO 0 AT R2. "	001240		3
1120				DHAZ - AZIMUTH ANGLE OF ATTACK OFF FRONT OF TARGET- TOWARD DRIVER SIDE. SET TO -1 FOR RANDOM- BLST - BLAST RADIUS WITHIN WHICH VEHICLE PK=PKBL- NOTE: TO ENTER BLAST RADII VS. BURST HEIGHT-.	001260		
1125				ENTER NEGATIVE NUMBER OF BLAST HGT PAIRS- IN PLACE OF VALUE OF BLST. PAIRS OF- BLAST.HGT ARE ENTERED IN ASCENDING ORDER- OF HEIGHT-.	001280		
1130			54	COORDINATE SYSTEM IS RECTANGULAR, " TARGET HEADING IS NEGATIVE RANGE, " WRITE (6,*), "DRIVER SIDE (LEFT) IS POSITIVE DEFLECTION, " WRITE (6,*), "HEIGHT IS MEASURED FROM GROUND" NPRT = 0 ISET = 0 ITIME = 0 CALL RDOUT(INIT) 15 CALL RDIN(INIT) ISET = 1	001300		
1135				IF(IRD.EQ.5) GO TO 88 IF(NPRT.GT.0) GO TO 80	001320		
1140				WRITE (6,*), "ENTER DATA BY ENTERING CODE NAME" WRITE (6,*), "FOLLOWED BY A COMMA AND THE VALUE IN FLOATING- WRITE (6,*), "PCNT FORMAT. TO END DATA ENTRY. ENTER " WRITE (6,*), "THE WORD ENC IN COLUMNS 1-3"	001340		
1145			C	C FILE TAPE1 CONTAINS BASIC INPUT DATA FILES TAPE2 - TAPE4 CONTAIN FRAGMENTATION PK GRIDS FOR DIFFERENT ANGLES OF ATTACK	001360		
1150			C	88 WRITE (6,*), "DO YOU WISH TO INITIALIZE DATA FROM" READ (5,1001) ANS	001380		
1155			C	IRD = 5 IF(ANS.EQ.YES) IRD = 1 80 REWIND 1 REWIND 2 REWIND 3 REWIND 4 PI = ATAN2(0.,-1.) DO 51 I=1,10 51 PKG(I) = 0.	001400		
1160			C	C INITIALIZE OR UPDATE DATA	001420		
1165			C	REWIND 1 7 IF(IRD.EQ.5) WRITE (6,*), "ENTER DATA OR END - " READ (IRD,1000) AAAA.VALUE 1000 FORMAT (A4,1X,F0.3) IF(AAAA.EQ.END) GO TO 14 DO 53 J=1,50 IF(AAAA.NE.ANAR(J)) GO TO 53 INEX(J) = 1 DO TO 2	001440		
1170			C	DATA(J) = VALUE 001460	001460		

PROGRAM ARP	73/74	OPT=1	FTN 4.8+508	03/13/81	08.28.23	PAGE	4
53	CONTINUE						
	WRITE (6,2000) AAAA						
	GO TO 7						
175	14	CALL READ (DATA,INew,ANAM,IPD,1,RDH,DDH,MDH)					
	C	SET UP TAPE1					
	C	9 REWIND 1					
180	DO 81 I=1,50						
	IF(DATA(I).EQ.0.) GO TO 81						
	WRITE (1,1000) ANAM(I),DATA(I)						
81	CONTINUE						
	C	WRITE (1,100C) END					
185	CALL WRITE (DATA,1,CEP,RDH,DDH,MDH)						
	C	REWIND 1					
		IF(ITIME.EQ.0.) GO TO 12					
		WRITE (6,*) "DO YOU WANT CURRENT INPUT LISTED?"					
		READ (5,1001) ANS					
190		IF(ANS.NE.YES) GO TO 23					
		IF(ITIME.GT.0) WRITE (6,*) "CURRENT DATA - "					
		12 IF(ITIME.EQ.0) WRITE (6,*) "INITIAL INPUTS - "					
195	C	LIST DATA FILE (TAPE1)					
	C	DO 8 I=1,50					
		READ (1,1000) A,B					
		IF(A,EQ.END) GO TO 6					
200	8 WRITE (6,1002) A,B						
	1002 FORMAT (1X,A4,1X,F10.3)						
	6 WRITE (6,1002) END						
		CALL WRITE (DATA,6,CEP,RDH,DDH,MDH)					
205	23 REWIND 1						
		ITIME = ITIME + 1					
		IF(ISET.EQ.1) GO TO 89					
		WRITE (6,*) "DO YOU WANT TO CHANGE ANY DATA? - "					
		READ (5,1001) ANS					
		IF(ANS.NE.YES) GO TO 82					
210	89 ISET = 0						
	C	READ IN CHANGES					
	C	DO 13 I=1,50					
215	13 INEW(I) = 0						
		DC 2 I=1,1000					
		WRITE (6,*) "ENTER DATA OR END - "					
		READ (5,100C) AAAA,VALUE					
		IF(AAAA.EQ.END) GO TO 3					
	1001 FORMAT (A1)						
		DC 4 J=1,50					
		IF(AAAA.NE.ANAM(J)) GO TO 4					
		DATA(J) = VALUE					
		INEW(J) = 1					
		GO TO 2					
220	4 CONTINUE						
		WRITE (6,2000) AAAA					
		2000 FORMAT (1X,"**** DC NOT RECOGNIZE ",A4," ****")					
225	2 CONTINUE						

PROGRAM ARP	73/74 OPT=1	FTN 4.8+508	03/13/81 08.2B.23	PAGE 5
230	3 CALL READ (DATA,INEW,ANAM,5,0,PDH,DOH,MDH) GO TO 9 82 DO 83 I=1,50 83 INEW(1) = 0	FZAM = DATA(1)/57.29578 FZTM = ABS(DATA(2)) PKDHX = DATA(3) PKBLX = DATA(4) FZAS = DATA(5)/57.29578 ITTG = 0 IF(FZAS.LT.0.) ITTG = 1 FZAS = ABS(FZAS)	002380 002390 002400 002410 002420 002430 002440 002450 002460 002470 002480 002490 002500 002510 002520 002530 002540 002550 002560 002570 002580 002590 002600 002610 002620 002630 002640 002650 002660 002670 002680 002690 002700 002710 002720 002730 002740 002750 002760 002770 002780 002790 002800 002810 002820 002830 002840 002850 002860 002870 002880 002890 002900 002910 002920 002930 CG2940	
235	C SET UP DATA C LOAD INPUT DATA INTO VARIABLE SET C AND CONVERT DEGREES TO RADIANS C	FZTS = DATA(6)/57.29578 CMEG = DATA(7)/57.29578 NGER = DATA(8) NCEP = DATA(9) IFUN = 0 NDHT = DATA(11) NSMP = DATA(14) NROR = DATA(15) DHAZ = DATA(16)/57.29578 NH = DATA(17) NA = 0 ONGS = 0. PKPF = DATA(21) NVT = DATA(20) NGLT = DATA(22) JGLT = 1 JGLT = ISIGN(JGLT,NGLT) NGLT = IABS(NGLT) SRNG = DATA(23) NPRT = DATA(24) NDEG = DATA(25) TGTC = DATA(26) DUDR = DATA(27) BLST = DATA(28) IF(BLST.LE.0.) GO TO 94 BBLST(1) = BLST HBLST(1) = 100000. BLST = 1. NBLST = ABS(BLST)		
240	FZAM = DATA(1)/57.29578 FZTM = ABS(DATA(2)) PKDHX = DATA(3) PKBLX = DATA(4) FZAS = DATA(5)/57.29578 ITTG = 0 IF(FZAS.LT.0.) ITTG = 1 FZAS = ABS(FZAS)	IMFZ = 0 IF(DATA(2).LT.0.) IMFZ = 1 IF(PKDHX.EQ.0.) PKDHX = 1. IF(PKBLX.EQ.0.) PKBLX = 1. NLQOP = NGER IF(NDBG.GE.1) WRITE (6,*)'DEBUG OPTION ',NDBG IF(DATA(2).NE.0.) IFU2 = 2 IF(DATA(1).NE.0.) IFU2 = 1 XRNG = 0. IF(NCHT.EQ.0) GO TO 115		
245				
250				
255				
260				
265				
270				
275				
280				
285				

PROGRAM ARP 73/74 OPT=1

FTN 4.8+508

8

PROGRAM ARP 73/74 CPT=1

FTN 4.8+508

PAGE 7

```
      BDBAR = 0.          003520  
      EDBAR2 = 0.        003530  
      BRBAR = 0.         003540  
      BREAR2 = 0.        003550  
      BHBAR = C.         003560  
      BHBAR2 = 0.        003570  
      IF(PKPF.EQ.0.) FKPF = 1.  
      IF(PKPF.LT.0.) PKPF = 0.  
      SIGD = SDD(ILUP)  
      SIGN = SDH(ILUP)  
      NCT = 0
```

355 C BEGIN SIMULATIONS
 C DO 1 ISIM=1,NSMP
 C IF(DATA(16).LT.0.) DHAZ = R2M(1)*2.*PI
 C PRSAMP = 0.0
 C PADH = 0.
 C PBLST = C.
 C PRDR = 0.
 C CHECK FOR DUO
 C IF(RDM(1).LE.DUDR) GO TO 18
 C C SAMPLE FROM ATTACK ANGLE DISTRIBUTION
45 376 C CALL BOXND (Z1,Z2)
 C QMOK = Z1*CGS + OMEG
 C SING = SIN(OMEGA)
 C COSG = CGS(OMEGA)
 C TANG = 1.
 C IF(COSG.NE.0.) TAND = SING/COSG
 C C ROTATE COORDINATES OF HAVING POINT ACCORDING
 C TC AZIMUTH COMPONENT OF ATTACK ANGLE.
 C ALL COMPUTATIONS TO DETERMINE FUZING POINT ARE IN
 C ROTATED COORDINATE SYSTEM.
 C GMRR = GMR
 C GMDR = GMD
 C CALL ROTATE (GMRR,GMDR,DHAZ,1.)
 C C SAMPLE FROM GUIDANCE ERROR DISTRIBUTION
 C RELATIVE TO HAVING POINT
 C CALL BOXND (G,H)
 C DMTN = SQRT ((SIGN*H)**2. + (SIGD*D)**2.)
 C GR = GMR + SIGN*H*SIGD
 C GD = GMDR + SIGD*D
 C GH = GMD + SIGN*H*COSG
 C C (GR,GD,GH) IS INTERCEPT OF
 C TRAJECTORY WITH GUIDANCE PLANE
 C (RF,DF,HF) WILL BE FUZING POINT ON TRAJECTORY.

PROGRAM APP 73/74 CPT=1 FTN 4.8+508 63/13/61 06.28.23 PAGE 6

```

400      C      RF = GR      004090
        DF = GO      004100
        HF = GH      004110
        C      CHECK FOR PRIMARY FUZE FUNCTION
        IBKUP = 0      004120
        IF(RDM(1).GT.PKFF) GC TO 16
        C      CHECK FOR HEIGHT FUZING
        IF(IFMFZ.EQ.1) GO TO 74
        D2 = 0.
        C      CHECK FOR APPROPRIATE FUZING
        CALL BXNC (Z1,Z2)
        IZO = IFUZ + 1
        IF(NDBG.GE.1) WRITE (6,503) IFUZ,IGO,GR,GC,CH
        GO TO 155,75,52,65,IGO
420      C      CROSS GLITTER POINT FOR FUZING, ANGULAR FUZE ONLY
        C      75 IF(JCLT.LT.G.AND.NGLT.GT.I) GO TO 76
        XGLT = NGLT
        IGLT = (RDM(1)-0.0001)*XGLT + 1.0
        IF(IGLT.EQ.0) IGLT + 1
        RGLT = GLTR(1,IGLT)
        GLT = GLTR(2,IGLT)
        HGLT = GLTR(3,IGLT)
        IGO = 1
        GO TO 77
        76 IGO = NGLT
        GRMAX = -100000.
        77 DO 82 IGL=1,100
        IF((ID0.EQ.1) GO TO 2)
        RGLT = GLTR(1,IGL)
        DGLT = GLTR(2,IGL)
        HGLT = GLTR(3,IGL)
        2: IF(NDBG.EQ.1) WRITE (6,*)
        *RGLT,DGLT,HGLT =
        *RGLT,DGLT,HGLT
        C      ROTATE GLITTER POINT INTO ARP COORDINATE SYSTEM
        CALL ROTATE (RGLT,DGLT,DHAZ,1.)
        IF(NDBG.EQ.1) WRITE (6,*)
        *ROTATED GLITTER POINT =
        IF(NDBG.EQ.1) WRITE (6,*)
        *DHAZ,RGLT,DGLT,HGLT =
        *DHAZ,RGLT,DGLT,HGLT
        5003 FORMAT (1X,*IFUZ,IGO = *,2(12,*,*,1X),*GR,GO,GH = *,3(F6.1,*,*,1X))
        C      USE LAW OF SINES AND LAW OF COSINES TO FIND
        FUZING POINT ON TRAJECTORY. FIRST PICK A POINT
        ALONG TRAJECTORY TO COMPUTE EtaX (ANGLE BETWEEN
        TRAJECTORY AND A LINE (AB) FROM GLITTER POINT
        (RGLT,DGLT,HGLT) TO GUIDANCE PLANE INTERCEPT
        (GE,GS,PH) = NOTE THAT EVERYTHING IS IN ROTATED
  
```

PROGRAM APP 73/74 GPT=1 FTN 4.8+508 03/13/81 08.28.23 PAGE 9

```

      C COORDINATE SYSTEM (THROUGH AZIMUTH ATTACK ANGLE
      C (ANG) COMPONENT). THEN, KNOWING BETAX AND FUZING ANGLE
      C (ANG), COMPUTE ANGLE (GAMMA) WITH ITS VERTEX AT
      C GLITTER POINT AND OPPOSITE TRAJECTORY SEGMENT
      C BOUNDED BY GUIDANCE PLANE INTERCEPT AND FUZING
      C POINT. FINALLY, KNOWING GAMMA, AB, AND ANG, COMPUTE
      C Q2, THE DISTANCE FROM GUIDANCE PLANE INTERCEPT
      C TO FUZING POINT (USING THE LAW OF SINES).
      C
      C TANGX = TANG
      C IF(SIND.EQ.0.) TANGX = 1.
      C CB = 16.
      C IF(SINC.NE.0.) CB = CB/SIND
      C
      C GRL, GRL, GHL ARE COORDINATES OF A POINT ON
      C THE TRAJECTORY USED TO COMPUTE BETAX.
      C
      C GRL = GR - 10./TANGX
      C GDL = GD
      C GHL = GH
      C IF(SIND.NE.0.) GHL = GH + 10.
      C A32 = (RGLT-CR)**2. + (DGLT-GD)**2. + (HGLT-GH)**2.
      C B32 = (RGLT-GRL)**2. + (DGLT-GDL)**2. + (HGLT-GHL)**2.
      C AB = SQRT(AB2)
      C
      C USE LAW OF COSINES TO COMPUTE BETAX, ANGLE WITH VERTEX AT
      C GLITTER POINT AND OPPOSITE TRAJECTORY SEGMENT
      C BOUNDED BY GUIDANCE PLANE INTERCEPT AND FUZING POINT.
      C
      C BETAX = ACOS((AB2-BB2+CB*CB)/(2.*AB*CB))
      C IF(NDBG.EQ.1) WRITE(6,*)
      C BETAX,GRL,GDL,GHL,AB,CB
      C FZASX = FZAS
      C IF(ITTG.EQ.1) FZASX = 0.
      C
      C ANGULAR FUZZING FUNCTION
      C
      C ANG = 22.*FZASX + FZAN
      C IF(FZAN.LT.0.) ANG = FZAN + RDH(1)*(FZASX-FZAN)
      C IF(ANG.LT.-0.1745) GO TO 18
      C IF(ANG.GT.PI) GO TO 16
      C
      C Q2 IS DISTANCE ALONG TRAJECTORY FROM GUIDANCE
      C PLANE INTERCEPT TO FUZING POINT.
      C
      C GAMMA = PI - BETAX - ANG
      C
      C IF GAMMA.LT.ZERO, USE SUPPLEMENT OF ANG FOR FUZING.
      C
      C IF(GAMMA.LT.C.) ANG = PI - ANG
      C Q2 = AB*(GIN(GAMMA)/SIN(ANG))
      C IF(NDBG.EQ.1) INFINITE(6,*) Q2,GAMMA,ANG = ,G2,GAMMA,ANG
      C IF(IGD.EQ.1) GO TO 22
      C IF(O2.LT.GRMAX) GO TO 84
      C GRMAX = Q2
      C IGLT = IGL
      C
      C 84 CONTINUE
  
```

PROGRAM ARP 73/74 OPT=1

FTN 4.8+508 03/13/81 08.28.23 PAGE 10

515 C GO TO 66
C C LINEAR FUZING FUNCTION (ALONG TRAJECTORY)
C C FUZING DIRECTION IS POSITIVE IN THE NEGATIVE
C C RANGE DIRECTION, I.E., A POSITIVE CHANGE IN
C C THE FUZING DISTANCE, D2, IS IN THE NEGATIVE
C C RANGE DIRECTION.
520 C 22 IF(ITS.EQ.1) FZTS = DMIN*TAN(FZAS)
C C D2 = D2 + Z2*FZTS + FZTM
C C RF = GR - D2*COSD
C C HF = GH + D2*SIND
C C DF = GD
C C GO TO 85
C C BACKUP FUZING
530 C 16 HF = 0.
C C 1BKUP = 1
C C IF(OMEGA.EQ.0.) GO TO 5
C C IF(NVT.EQ.0.) GO TO 17
535 87 XK = RDML1
C C DO 65 K=1,NVT
C C KK = K
C C IF(XK.LE.PVT(K)) GO TO 66
C C 65 CONTINUE
C C 66 HFX = VHT(KK)
C C IF(HFX.LE.HF) GO TO 24
C C HF = HFX
C C 17 RF = GR - (HF-GH)/TAND
C C DF = GD
C C GO TO 61
540 48 5 WRITE(6,*), "NO BACKUP FUZING FOR OMEGA = 0."
C C WRITE(6,*), "TRAJECTORY CLOSEST POINT OF APPROACH TO TARGET"
C C WRITE(6,*), "CENTER IS USED"
C C RF = 0.
C C DF = GD
C C HF = GH
C C GO TO 61
C C HEIGHT FUZING
555 C 55 IF(SIND.EQ.0.) STCP 74
C C CALL BXND(Z1,Z2)
C C HF = FZTM + Z1*FZTS
C C RF = RF + (GH-HF)/TAND
C C 85 IF(OMEGA.EQ.0.) GO TO 24
C C IF(NVT.NE.0) GO TO 87
C C C CHECK FOR FUZING POINT BELOW GROUND
C C 24 IF(HF.GE.0.) GO TO 61
C C IF(OMEGA.EQ.0.) GO TO 61
C C RF = RF + HF/TAND
C C HF = 0.
560 C C PUT BURST POINT IN TARGET COORDINATE SYSTEM FOR
570 C 005230
005240
005250
005260
005270
005280
005290
005300
005310
005320
005330
005340
005350
005360
005370
005380
005390
005400
005410
005420
005430
005440
005450
005460
005470
005480
005490
005500
005510
005520
005530
005540
005550
005560
005570
005580
005590
005600
005610
005620
005630
005640
005650
005660
005670
005680
005690
005700
005710
005720
005730
005740
005750
005760
005770
005780
005790

PROGRAM ARP	73/74	OPT=1	FTN 4.8+508	03/13/81	08.26.23
	C	BLAST AND DIRECT HIT COMPUTATIONS.			
	C	61 CALL ROTATE (RF,DF,DHAZ,-1.)			
	C	LR = RF	005800	005810	
	C	BD = DF	005820	005830	
	C	BH = HF	005840	005850	
	C	IF(NDBG.GE.1) WRITE (6,*),*BR,BD,BH AT SINT 61 = *,BR,BD,BH	005860	005870	
575	C	SET UP BLST VALUE FOR BLST VS. HGT	005880	005890	
	C	IF(NBLST.LE.0) GO TO 105	005900	005910	
	C	DO 10 I=1,NBLST	005920	005930	
	C	IF(HF.GT.HBLST(I)) GO TO 10	005940	005950	
	C	BLST = BBLST(I)	005960	005970	
	C	GO TO 105	005980	005990	
585	C	10 CONTINUE	006000	006010	
	C	BLST = 0.	006020	006030	
	C	WRITE (6,*),*HF EXCEEDS ALL HBLST, HF = *,HF	006040	006050	
	C	GO TO 18	006060	006070	
590	C	105 IF(NDHT.EQ.0) GO TO 106	006080	006090	
	C	DETERMINE DIRECT HIT PK	006100	006110	
	C	USE 2 POINTS TO DEFINE TRAJECTORY, BURST POINT (BR,BD,BH) AND POINT AT BR+10 (RBS,DBS,HBS). IF AZIMUTH ATTACK ANGLE IS 90 DEGREES, SET RBS,DBS,HBS POINT AT ED+10. (RPN,DPN,HPN) WILL BE BURST POINT, WITH OR WITHOUT DIRECT HIT.	006120	006130	
600	C	IPN IS PENETRATION INDEX (0 = NO PENETRATION, N = BOX N PENETRATED)	006140	006150	
	C	RPN = BR	006160	006170	
	C	DPN = BD	006180	006190	
	C	HPN = BH	006200	006210	
	C	IF(ABS(DATA(16)).EQ.90.) GO TO 95	006220	006230	
	C	RBS = BR + 1C.	006240	006250	
	C	DBS = BD - 1C.*TAN(DHAZ)	006260	006270	
	C	HBS = BH - 1C.*TANO/COS(DHAZ)	006280	006290	
	C	GO TO 96	006300	006310	
	C	95 RBS = BR	006320	006330	
	C	DBS = BD + 10.	006340	006350	
	C	HBS = BH + 10.*TANO	006360	006370	
615	C	96 IPN = 0	006380	006390	
	C	CHECK EACH BOX FOR PENETRATION	006400	006410	
	C	IF(NDBG.EQ.1) WRITE (6,*),*OMEGA,RBS,DBS,HBS = ,OMEGA,RBS,DBS,DBS,HBS	006420	006430	
620	C	IF(NDBG.EQ.1) WRITE (6,*),*PF,DF,HF = ,RF,DF,HF	006440	006450	
	C	IF(NDBG.EQ.1) WRITE (6,*),*GR,GD,GH = ,GR,GD,GH	006460	006470	
	C	DO 92 I=1,NDHT	006480	006490	
	C	IF(BR.LT.RDH(I,1)) GO TO 92	006500	006510	
	C	IF(BD.LT.RDH(I,1).NE.0) GO TO 109	006520	006530	
	C	IF(BD.LT.DDH(I,1).OR.BD.GT.DDH(I,2)) GO TO 92	006540	006550	
	C	IF(BH.GT.HDH(I,2).AND.OMEGA.GE.0.) GO TO 92	006560	006570	
625	C	109	006580	006590	

FTN 4.8+508

PROGRAM ARP	73/74	OPT=1	03/13/81	08.28.23	PAGE	
					12	
630	C	IF(BH.LT.HDH(I,1)).AND.OMEGA.EQ.0.) GO TO 92 RDH1 = RDH(1,1) RDH2 = RDH(1,2) DDH1 = DDH(1,1) DDH2 = DDH(1,2) HDH1 = HDH(1,1) HDH2 = HDH(1,2)	006370 006380 006390 006400 006410 006420 006430 006440			
635	C	IPEN = NUMBER OF SIDES PENETRATED (MUST BE 0 OR 2) C IPEN = 0 IF(ABS(DATA(16)).EQ.90.) GO TO 102	006450 006460 006470 006480 006490			
640	C	C CHECK RANGE SIDES DO 97 K=1,2 RDHX = RDH1 IF(K.EQ.2) RDHX = RDH2 CALL SEARCH (I,1,RDHX,DA,HA) IF(NDBG.EQ.2) WRITE (6,*) "IPEN, RDHX, DA, HA = ", IPEN, 1 RDHX, DA, HA 97 CONTINUE IF(IPEN.EQ.2) GO TO 92 102 IF(DATA(16).EQ.0.OR.DATA(16).EQ.180.) GO TO 108	006500 006510 006520 006530 006540 006550 006560 006570 006580 006590			
645	C	C CHECK DEFLECTION SIDES DO 107 K=1,2 DDHX = DDH1 IF(K.EQ.2) DDXH = DDXH2 CALL SEARCH (I,2,RA,DDHX,HA) IF(NDBG.EQ.2) WRITE (6,*) "IPEN, RA, DDXH, HA = ", IPEN, 1 RA,DDHX,HA 107 CONTINUE 108 IF(OMEGA.EQ.0.) GO TO 101	006600 006610 006620 006630 006640 006650 006660 006670 006680 006690 006700 006710 006720 006730 006740			
650	C	C CHECK HEIGHT SIDES DO 117 K=1,2 HDHX = HDH1 IF(K.EQ.2) HDHX = HDH2 CALL SEARCH (I,3,RA,HDHX,HA) IF(NDBG.EQ.2) WRITE (6,*) "IPEN, RA, DA, HDHX = ", IPEN, 1 RA,DA,HDHX 117 CONTINUE 101 IF(IPEN.EQ.1) STCP 117 92 CONTINUE IF(IPEN.EQ.0) GO TO 106 PKDH = PKDH + PKDH	006750 006760 006770 006780 006790 006800 006810 006820 006830 006840 006850 006860 006870 006880 006890 006900 006910 006920 006930			
655	C	50				
660	C					
665	C					
670	C					
675	C					
680	C	SET UP BURST COORDINATES (BR,BD,BH) FROM DIRECT HIT. BR = RP BD = DP BH = HP				

	PROGRAM ARP	73/74 OPT=1	FTN 4.8+508	03/13/81 08.28.23	PAGE 13
685	106 IF(BH.GE.0.) GO TO 37 IF(OMEGA.EQ.0.) STOP 106 BR = BR + BH/TANQ BH = 0.			006940 006950 006960 006970 006980 006990 007000 007010 007020 007030 007040 007050 007060 007070 007080 007090 007100 007110 007120 007130 007140 007150 007160 007170 007180 007190 007200 007210 007220 007230 007240 007250 007260 007270 007280 007290 007300 007310 007320 007330 007340 007350 007360 007370 007380 007390 007400 007410 007420 007430 007440 007450 007460 007470 007480 007490 007500	
690	C COMPUTE NEAR MISS BLAST KILL				
695	C 37 IF(NBLST.EQ.0) GO TO 90 IF(NCHT.EQ.0) GO TO 103 DO 104 I=1,NDHT IBLST = 1 CALL BLAST (IBLST,BR,BLST,RCH,I) CALL BLAST (IBLST,BD,BLST,HDH,I) CALL BLAST (IBLST,BH,BLST,HDH,I) IF(IBLST.EQ.1) GO TO 11				
700	104 CONTINUE GO TO 90				
705	103 DIST = SQRT(BR*BR + BD*BD + (BH-TGTC)*(BH-TGTC)) IF(DIST.GT.BLST) GO TO 90 11 PKBLST = PKBLST + PKBLX				
710	C COMPUTE RADAR BLAST KILL				
715	C 90 IF(NOBG.EQ.2) WRITE (6,*)'IPN,RPN,DPN,HPN,BR,BD,BH = '. C IPN,RPN,DPN,HPN,BR,BD,BH IF(NRDR.EQ.0) GO TO 27 NRDR = BR-RDR(1) RDR = BD-RDR(2) RDR = BH-RDR(3) RDR = SORT(BRDR-BDR+DRDR*DRDR+HRDR+HRDR) PKRDR = 1.0 IF(RRDR.GT.RDR(4)) PKRDR = 1. - (RRDR-RDR(4))/(RDR(5)-RDR(4)) IF(RRDR.GE.RDR(5)) PKRDR = 0. 5004 FORMAT (1X,*BR,ED,BH = *,3(F6.1,*,*1X)) 27 IBX = 0 IROT = 0 IF(NDBG.GE.1) WRITE (6,5004) BR,BD,BH IF(NH.EQ.0) GO TO 50				
720	C COMPUTE PK DUE TO FRAGMENTATION (PKSAMP)				
725	C C INTERPOLATE IN RANGE, DEFLECTION, HEIGHT & ANGLE TO C GET FRAGMENTATION PK FROM PK GRIDS.				
730	C C II = 1 IF(BH.GT.HGT(NH+1)) GO TO 50 C ROTATE BURST POINT FOR FRAGMENTATION PK C INTERPOLATION INTO ARP COORDINATE SYSTEM. C RECALL THAT PK GRIDS ARE IN PROJECTILE COORDINATE C SYSTEM.				
735	C C CALL ROTATE (BR,BD,DHAZ,1.) IROT = 1 C LOCATE HEIGHT BOUNDARIES				
740	C				

PROGRAM ARP 73/74 OPT=1 FTN 4.8+508 03/13/81 08.28.23 PAGE 14

```

C      DC 20 I=1,NH          007510
      IH2 = 1                 007520
      IF(BH.LE.HGT(I)) GO TO 25
      20 CONTINUE               007530
      IH2 = 0                 007540
      25 IH1 = IH2 - 1         007550
      IF(IH1.EQ.0) IH1 = 1     007560
      IF(IH1.LT.0) IH1 = NH    007570
      IF(NDBG.EQ.4) WRITE(6,*),IH1,IH2,NR,ND,RU,DJ,BR,BD,BH = *
      C IH1,IH2,NR,ND,RU,DJ,BR,BD,BH
      31 CALL INTERP(BR,BD,BH,RORD,CGRD,HGT,IH1,IH2,PK1,PKA,NR,ND,RU,DJ,NH,007610
      C NDBG)
      PKSAMIP = PKA            007620
      GO TO 41                 007630
      50 PKSAMIP = 0.           007640
      50 PKSAMIP = 0.           007650
      50 PKSAMIP = 0.           007660
      50 PKSAMIP = 0.           007670
      C COMPUTE SPHERICAL COORDINATES TO BURST POINT (BR,BD,BH)
      C FROM GROUND ZERO (0,0,0)                         007680
      C SAI = ANGLE OFF POSITIVE RANGE AXIS MEASURED
      C CLOCKWISE                                         007690
      C SA2 = ANGLE OFF R-D PLANE MEASURED TOWARD POSITIVE
      C SR = RANGE FROM BURST POINT TO (0,0,0)           007700
      C H-AXIS IN VERTICLE PLANE                         007710
      C
      C 41 IF(NDBG.EQ.4) WRITE(6,*),PK(FRAG) = *,PKSAMP
      C GET BURST POINT BACK INTO TARGET COORDINATE
      C SYSTEM IF IROT = 1.                                007720
      C
      C IF(IROT.EQ.1) CALL ROTATE (ER,BD,DHAZ,-1.)
      BRR = BR*BR
      BDD = BD*BD
      BH = BH*BH
      RRR = BRR + BDD + BH
      RR = SQRT(RRR)
      WRITE(6,*),ER,BD,BH,RR
      BRBAR = BRBAR + ER
      BRBAR2 = BRBAR2 + BRR
      BDBAR = BD*BAR + BD
      BDBAR2 = BDBAR2 + BDD
      BHBAR = BH*BAR + BH
      BHBAR2 = BHBAR2 + BHH
      RRBAR = RRBAR + RR
      RRBAR2 = RRBAR2 + RRR
      SA1 = PI/2.
      SA2 = 0.
      IF(BR.EQ.0.) GO TO 55
      SA1 = ATAN2(BD,BR)
      IF(SA1.LT.0) SA1 = 2.*PI + SA1
      55 IF(BD.EQ.0. .AND.BR.EQ.0.) GO TO 56
      SA2 = ATAN(BH/SQRT(BR*BR+BD*BD))
      56 SA1 = SA1*360./{(2.*PI)
      SA2 = SA2*360./{(2.*PI)
      DO 57 I=1,12
      ISAI = 1
      IF(SA1.LT.ALPHA(I+1)) GO TO 58
  
```

FTN 4.8+508 03/13/81 08.28.23 PAGE 15

```

PROGRAM ARP      73/74   OPT=1

      57 CONTINUE
      58 DO 98 I=1,6
      59   ISA2 = 1
      60   IF(ISA2.LT.BETA(I+1)) GO TO 99
      61 CONTINUE
      62 SR = SQRT(BR*BR + BD*BD + SH*BH)
      63 ISR = 0
      64 DO 100 I=1,10
      65   II = I
      66   IF(I.EQ.10) II = 11
      67   ISR = ISR + 1
      68   IF(SF.LT.RANGE(II)) GO TO 110
      69   IF(NDBG.EQ.6) WRITE(6,*),ISA1,ISA2,ISR = " ,ISA1,ISA2,ISR
      70   IF(NDBG.EQ.6) WRITE(6,*),SA1,SA2,SR = " ,SA1,SA2,SR
      71
      C STORE PK'S ACCORDING TO SPHERICAL COORDINATES
      72
      C IKS(ISA1,ISA2,ISR) = IKS(ISA1,ISA2,ISR) + 1
      73
      C SUM PK'S OVER ALL SAMPLES
      74
      C IF(NDBG.GT.0) WRITE(6,*) "PKR,PKR,PKD,PKB = ",PKSAM,PKRD,PKDR
      75   C PKBLST
      76   PKBASE = PKBASE + PKSAM
      77   PKRADR = PKRADR + PKDR
      78   PKDHIT = PKDHIT + PKDH
      79   PKBLT = PKBLT + PKBLST
      80   PKSAM = 1. - (1.-PKSAM)*(1.-PKDH)*(1.-PKBLST)
      81   PKSI(ISA1,ISA2,ISR) = PKS(ISA1,ISA2,ISR) + PKSAM
      82   PKTOT = PKTOT + PKSAM
      83   PKTOT2 = PKTOT2 + PKSAM*PKSAM
      84   IF(NDBG.GE.1) WRITE(6,3003) PKSAM
      85   3003 FORMAT(5X,"SAMPLE PK = *,F6.4")
      86   IF(INPRT.EQ.1) GO TO 1
      87   IF(MOD(ISIM,10).NE.0) GO TO 1
      88   PKPRNT = ISIM
      89   PKPRNT = PKTOT/PKPRNT
      90   WRITE(6,*),"NO. SIMULATIONS, PK = ",ISIM,PKPRNT
      91   GO TO 1
      92   18 NCT = NCT + 1
      93   CONTINUE
      94
      C DISPLAY FINAL RESULTS
      95
      C IF(INPRT.GT.0) GO TO 79
      96   WRITE(6,2002)
      97   WRITE(6,*),"FINAL RESULTS"
      98   79  XSAM = NSMP
      99   FORMAT(/,1X,*PK = *,F6.4,2X,*PKSD = *,F6.4,2X,*NSAMP = *,16,/)

      8000 PKBAR = PKTOT/XSAM
      8001 PKBASE = PKBASE/XSAM
      8002 PKRADR = PKRADR/XSAM
      8003 PKDHIT = PKDHIT/XSAM
      8004 PKBLT = PKBLT/XSAM
      8005 PK(1LUP) = PKBASE
      8006 PKR(1LUP) = PKRADR
      8007
      8008
      8009
      8010
      8011
      8012
      8013
      8014
      8015
      8016
      8017
      8018
      8019
      8020
      8021
      8022
      8023
      8024
      8025
      8026
      8027
      8028
      8029
      8030
      8031
      8032
      8033
      8034
      8035
      8036
      8037
      8038
      8039
      8040
      8041
      8042
      8043
      8044
      8045
      8046
      8047
      8048
      8049
      8050
      8051
      8052
      8053
      8054
      8055
      8056
      8057
      8058
      8059
      8060
      8061
      8062
      8063
      8064
      8065
      8066
      8067
      8068
      8069
      8070
      8071
      8072
      8073
      8074
      8075
      8076
      8077
      8078
      8079
      8080
      8081
      8082
      8083
      8084
      8085
      8086
      8087
      8088
      8089
      8090
      8091
      8092
      8093
      8094
      8095
      8096
      8097
      8098
      8099
      8100
      8101
      8102
      8103
      8104
      8105
      8106
      8107
      8108
      8109
      8110
      8111
      8112
      8113
      8114
      8115
      8116
      8117
      8118
      8119
      8120
      8121
      8122
      8123
      8124
      8125
      8126
      8127
      8128
      8129
      8130
      8131
      8132
      8133
      8134
      8135
      8136
      8137
      8138
      8139
      8140
      8141
      8142
      8143
      8144
      8145
      8146
      8147
      8148
      8149
      8150
      8151
      8152
      8153
      8154
      8155
      8156
      8157
      8158
      8159
      8160
      8161
      8162
      8163
      8164
      8165
      8166
      8167
      8168
      8169
      8170
      8171
      8172
      8173
      8174
      8175
      8176
      8177
      8178
      8179
      8180
      8181
      8182
      8183
      8184
      8185
      8186
      8187
      8188
      8189
      8190
      8191
      8192
      8193
      8194
      8195
      8196
      8197
      8198
      8199
      8200
      8201
      8202
      8203
      8204
      8205
      8206
      8207
      8208
      8209
      8210
      8211
      8212
      8213
      8214
      8215
      8216
      8217
      8218
      8219
      8220
      8221
      8222
      8223
      8224
      8225
      8226
      8227
      8228
      8229
      8230
      8231
      8232
      8233
      8234
      8235
      8236
      8237
      8238
      8239
      8240
      8241
      8242
      8243
      8244
      8245
      8246
      8247
      8248
      8249
      8250
      8251
      8252
      8253
      8254
      8255
      8256
      8257
      8258
      8259
      8260
      8261
      8262
      8263
      8264
      8265
      8266
      8267
      8268
      8269
      8270
      8271
      8272
      8273
      8274
      8275
      8276
      8277
      8278
      8279
      8280
      8281
      8282
      8283
      8284
      8285
      8286
      8287
      8288
      8289
      8290
      8291
      8292
      8293
      8294
      8295
      8296
      8297
      8298
      8299
      8300
      8301
      8302
      8303
      8304
      8305
      8306
      8307
      8308
      8309
      8310
      8311
      8312
      8313
      8314
      8315
      8316
      8317
      8318
      8319
      8320
      8321
      8322
      8323
      8324
      8325
      8326
      8327
      8328
      8329
      8330
      8331
      8332
      8333
      8334
      8335
      8336
      8337
      8338
      8339
      8340
      8341
      8342
      8343
      8344
      8345
      8346
      8347
      8348
      8349
      8350
      8351
      8352
      8353
      8354
      8355
      8356
      8357
      8358
      8359
      8360
      8361
      8362
      8363
      8364
      8365
      8366
      8367
      8368
      8369
      8370
      8371
      8372
      8373
      8374
      8375
      8376
      8377
      8378
      8379
      8380
      8381
      8382
      8383
      8384
      8385
      8386
      8387
      8388
      8389
      8390
      8391
      8392
      8393
      8394
      8395
      8396
      8397
      8398
      8399
      8400
      8401
      8402
      8403
      8404
      8405
      8406
      8407
      8408
      8409
      8410
      8411
      8412
      8413
      8414
      8415
      8416
      8417
      8418
      8419
      8420
      8421
      8422
      8423
      8424
      8425
      8426
      8427
      8428
      8429
      8430
      8431
      8432
      8433
      8434
      8435
      8436
      8437
      8438
      8439
      8440
      8441
      8442
      8443
      8444
      8445
      8446
      8447
      8448
      8449
      8450
      8451
      8452
      8453
      8454
      8455
      8456
      8457
      8458
      8459
      8460
      8461
      8462
      8463
      8464
      8465
      8466
      8467
      8468
      8469
      8470
      8471
      8472
      8473
      8474
      8475
      8476
      8477
      8478
      8479
      8480
      8481
      8482
      8483
      8484
      8485
      8486
      8487
      8488
      8489
      8490
      8491
      8492
      8493
      8494
      8495
      8496
      8497
      8498
      8499
      8500
      8501
      8502
      8503
      8504
      8505
      8506
      8507
      8508
      8509
      8510
      8511
      8512
      8513
      8514
      8515
      8516
      8517
      8518
      8519
      8520
      8521
      8522
      8523
      8524
      8525
      8526
      8527
      8528
      8529
      8530
      8531
      8532
      8533
      8534
      8535
      8536
      8537
      8538
      8539
      8540
      8541
      8542
      8543
      8544
      8545
      8546
      8547
      8548
      8549
      8550
      8551
      8552
      8553
      8554
      8555
      8556
      8557
      8558
      8559
      8560
      8561
      8562
      8563
      8564
      8565
      8566
      8567
      8568
      8569
      8570
      8571
      8572
      8573
      8574
      8575
      8576
      8577
      8578
      8579
      8580
      8581
      8582
      8583
      8584
      8585
      8586
      8587
      8588
      8589
      8590
      8591
      8592
      8593
      8594
      8595
      8596
      8597
      8598
      8599
      8600
      8601
      8602
      8603
      8604
      8605
      8606
      8607
      8608
      8609
      8610
      8611
      8612
      8613
      8614
      8615
      8616
      8617
      8618
      8619
      8620
      8621
      8622
      8623
      8624
      8625
      8626
      8627
      8628
      8629
      8630
      8631
      8632
      8633
      8634
      8635
      8636
      8637
      8638
      8639
      8640
      8641
      8642
      8643
      8644
      8645
      8646
      8647
      8648
      8649
      8650
      8651
      8652
      8653
      8654
      8655
      8656
      8657
      8658
      8659
      8660
      8661
      8662
      8663
      8664
      8665
      8666
      8667
      8668
      8669
      8670
      8671
      8672
      8673
      8674
      8675
      8676
      8677
      8678
      8679
      8680
      8681
      8682
      8683
      8684
      8685
      8686
      8687
      8688
      8689
      8690
      8691
      8692
      8693
      8694
      8695
      8696
      8697
      8698
      8699
      8700
      8701
      8702
      8703
      8704
      8705
      8706
      8707
      8708
      8709
      8710
      8711
      8712
      8713
      8714
      8715
      8716
      8717
      8718
      8719
      8720
      8721
      8722
      8723
      8724
      8725
      8726
      8727
      8728
      8729
      8730
      8731
      8732
      8733
      8734
      8735
      8736
      8737
      8738
      8739
      8740
      8741
      8742
      8743
      8744
      8745
      8746
      8747
      8748
      8749
      8750
      8751
      8752
      8753
      8754
      8755
      8756
      8757
      8758
      8759
      8760
      8761
      8762
      8763
      8764
      8765
      8766
      8767
      8768
      8769
      8770
      8771
      8772
      8773
      8774
      8775
      8776
      8777
      8778
      8779
      8780
      8781
      8782
      8783
      8784
      8785
      8786
      8787
      8788
      8789
      8790
      8791
      8792
      8793
      8794
      8795
      8796
      8797
      8798
      8799
      8800
      8801
      8802
      8803
      8804
      8805
      8806
      8807
      8808
      8809
      8810
      8811
      8812
      8813
      8814
      8815
      8816
      8817
      8818
      8819
      8820
      8821
      8822
      8823
      8824
      8825
      8826
      8827
      8828
      8829
      8830
      8831
      8832
      8833
      8834
      8835
      8836
      8837
      8838
      8839
      8840
      8841
      8842
      8843
      8844
      8845
      8846
      8847
      8848
      8849
      8850
      8851
      8852
      8853
      8854
      8855
      8856
      8857
      8858
      8859
      8860
      8861
      8862
      8863
      8864
      8865
      8866
      8867
      8868
      8869
      8870
      8871
      8872
      8873
      8874
      8875
      8876
      8877
      8878
      8879
      8880
      8881
      8882
      8883
      8884
      8885
      8886
      8887
      8888
      8889
      8890
      8891
      8892
      8893
      8894
      8895
      8896
      8897
      8898
      8899
      8900
      8901
      8902
      8903
      8904
      8905
      8906
      8907
      8908
      8909
      8910
      8911
      8912
      8913
      8914
      8915
      8916
      8917
      8918
      8919
      8920
      8921
      8922
      8923
      8924
      8925
      8926
      8927
      8928
      8929
      8930
      8931
      8932
      8933
      8934
      8935
      8936
      8937
      8938
      8939
      8940
      8941
      8942
      8943
      8944
      8945
      8946
      8947
      8948
      8949
      8950
      8951
      8952
      8953
      8954
      8955
      8956
      8957
      8958
      8959
      8960
      8961
      8962
      8963
      8964
      8965
      8966
      8967
      8968
      8969
      8970
      8971
      8972
      8973
      8974
      8975
      8976
      8977
      8978
      8979
      8980
      8981
      8982
      8983
      8984
      8985
      8986
      8987
      8988
      8989
      8990
      8991
      8992
      8993
      8994
      8995
      8996
      8997
      8998
      8999
      9000
      9001
      9002
      9003
      9004
      9005
      9006
      9007
      9008
      9009
      9010
      9011
      9012
      9013
      9014
      9015
      9016
      9017
      9018
      9019
      9020
      9021
      9022
      9023
      9024
      9025
      9026
      9027
      9028
      9029
      9030
      9031
      9032
      9033
      9034
      9035
      9036
      9037
      9038
      9039
      9040
      9041
      9042
      9043
      9044
      9045
      9046
      9047
      9048
      9049
      9050
      9051
      9052
      9053
      9054
      9055
      9056
      9057
      9058
      9059
      9060
      9061
      9062
      9063
      9064
      9065
      9066
      9067
      9068
      9069
      9070
      9071
      9072
      9073
      9074
      9075
      9076
      9077
      9078
      9079
      9080
      9081
      9082
      9083
      9084
      9085
      9086
      9087
      9088
      9089
      9090
      9091
      9092
      9093
      9094
      9095
      9096
      9097
      9098
      9099
      9100
      9101
      9102
      9103
      9104
      9105
      9106
      9107
      9108
      9109
      9110
      9111
      9112
      9113
      9114
      9115
      9116
      9117
      9118
      9119
      9120
      9121
      9122
      9123
      9124
      9125
      9126
      9127
      9128
      9129
      9130
      9131
      9132
      9133
      9134
      9135
      9136
      9137
      9138
      9139
      9140
      9141
      9142
      9143
      9144
      9145
      9146
      9147
      9148
      9149
      9150
      9151
      9152
      9153
      9154
      9155
      9156
      9157
      9158
      9159
      9160
      9161
      9162
      9163
      9164
      9165
      9166
      9167
      9168
      9169
      9170
      9171
      9172
      9173
      9174
      9175
      9176
      9177
      9178
      9179
      9180
      9181
      9182
      9183
      9184
      9185
      9186
      9187
      9188
      9189
      9190
      9191
      9192
      9193
      9194
      9195
      9196
      9197
      9198
      9199
      9200
      9201
      9202
      9203
      9204
      9205
      9206
      9207
      9208
      9209
      9210
      9211
      9212
      9213
      9214
      9215
      9216
      9217
      9218
      9219
      9220
      9221
      9222
      9223
      9224
      9225
      9226
      9227
      9228
      9229
      9230
      9231
      9232
      9233
      9234
      9235
      9236
      9237
      9238
      9239
      9240
      9241
      9242
      9243
      9244
      9245
      9246
      9247
      9248
      9249
      9250
      9251
      9252
      9253
      9254
      9255
      9256
      9257
      9258
      9259
      9260
      9261
      9262
      9263
      9264
      9265
      9266
      9267
      9268
      9269
      9270
      9271
      9272
      9273
      9274
      9275
      9276
      9277
      9278
      9279
      9280
      9281
      9282
      9283
      9284
      9285
      9286
      9287
      9288
      9289
      9290
      9291
      9292
      9293
      9294
      9295
      9296
      9297
      9298
      9299
      9300
      9301
      9302
      9303
      9304
      9305
      9306
      9307
      9308
      9309
      9310
      9311
      9312
      9313
      9314
      9315
      9316
      9317
      9318
      9319
      9320
      9321
      9322
      9323
      9324
      9325
      9326
      9327
      9328
      9329
      9330
      9331
      9332
      9333
      9334
      9335
      9336
      9337
      9338
      9339
      9340
      9341
      9342
      9343
      9344
      9345
      9346
      9347
      9348
      9349
      9350
      9351
      9352
      9353
      9354
      9355
      9356
      9357
      9358
      9359
      9360
      9361
      9362
      9363
      9364
      9365
      9366
      9367
      9368
      9369
      9370
      9371
      9372
      9373
      9374
      9375
      9376
      9377
      937
```

```

PROGRAM ARP    73/74   QFT=1          FTN 4.8+508   03/13/81  08.28.23   PAGE
1      PKD(ILUP) = PKDHT
2      PKG(ILUP) = PKBAR
3      PKBL(ILUP) = PKBLT
4      XSAMP = NSMP - NCT
5      IF(NSMP.EQ.NCT) XSAMP = 1.
6      XSMP = XSAMP-1.
7      IF(XSMP.EQ.0.) XSMP = 1.
8      BRG(ILUP) = BRBAR/XSAMPA
9      BOD(ILUP) = BODBAR/XSAMPA
10     BHG(ILUP) = BHGBAR/XSAMPA
11     RRG(ILUP) = PRBAR/XSAMPA
12     BRSG(ILUP) = SQRT((BRBAR2 - XSAMP*BRG(ILUP))*BRG(ILUP))/XSMP
13     BDSG(ILUP) = SQRT((BDBAR2 - XSAMP*BOD(ILUP))/XSMP)
14     BHSG(ILUP) = SQRT((BBAR2 - XSAMP*BHG(ILUP))/XSMP)
15     RRSG(ILUP) = SQRT((RBAR2 - XSAMP*RRG(ILUP))/XSMP)
16     IF(NCT.NE.0) WRITE(6,2004) NCT,NSMP,CEP(ILUP)
17     2004 FORMAT(2X,*PROJECTILE OR FUZING DUDS = *14,* OUT OF *,14,
18     C * SIMULATIONS,*./2X,* GUIDANCE CED = *,F6.2)
19     IF(INPRT.GT.0) GC TO 69
20     PKSD = (PK(OT2 - XSAMP*PKBAR*PKBAR))/XSMP
21     IF(PKSD.LT.0.) PKSD = 0.
22     PKSD = SQRT(PKSD)
23     WRITE(6,3000) PKBAR,PKSD,NSMP
24     WRITE(6,2002)
25     WRITE(6,*) "DO YOU WANT PK VS R. ALPHA, BETA? "
26     READ(5,1001) ANS
27     IF(ANS.NE.YES) GO TO 44
28
29     C
30     C   PK VS R, ALPHA, BETA, WHERE ALPHA IS AZIMUTH ANGLE
31     C   MEASURED FROM POSITIVE RANGE AXIS TOWARD POSITIVE
32     C   DEFLECTION AXIS (0 TO 360), BETA IS ELEVATION ANGLE
33     C   MEASURED FROM NEGATIVE HEIGHT AXIS TO POSITIVE
34     C   HEIGHT AXIS (0 TO 90).
35     C
36     WRITE(6,2001)
37     WRITE(6,*) PK          R           ALPHA          BETA
38     WRITE(6,*) -----  -----  -----  -----
39     DO 49 I=1,10
40     DO 49 J=1,12
41     DO 49 K=1,6
42     IF(IKS(J,K,I).EQ.0) GO TO 49
43     XIKS = IKS(J,K,I)
44     PKS(J,K,I) = PKS(J,K,I)/XIKS
45     CONTINUE
46     DO 45 I=1,10
47     XI = 0.
48     RSUM(I) = 0.
49     RANG = RANGE(I)
50     DO 47 J=1,12
51     DO 47 K=1,6
52     RPK = PKS(J,K,I)
53     XIKS = IKS(J,K,I)
54     XI = XI + XIKS
55     FSUM(I) = RSUM(I) + XIKS*RPK
56     IF(RPK.GT.0.) WRITE(6,3004) RPK,RANG,ALPHA(J),BETA(K),CBEATA(K+1)
57
58     3004 FORMAT(1X F6.4,2X,F5.1,2(2X F6.1,* - *,F6.1))

```

PROGRAM ARP

73/74 OPT=1

FTN 4.8+508

03/13/81 08.2B.23

PAGE 17

```

47 CONTINUE
IF(XI.EQ.0.) GO TO 45
RSUM(I) = RSUM(I)/XI
45 CONTINUE
WRITE (6,2002)
WRITE (6,* ) "AVG PK VS. R"
WRITE (6,* )
DO 43 I=1,10
R = RANGE(I)
IF(RSUM(I).EQ.0.) GO TO 43
WRITE (6,3001) RSUM(I),R
43 CONTINUE
3001 FORMAT (1X,F6.4,X,F5.1)
WRITE (6,2502)
C CHECK FOR ANOTHER CASE
C
C 44 WRITE (6,2001)
69 CONTINUE
C
C DISPLAY RESULTS FOR EACH GUIDANCE ERROR
C
IINPRT GT 0) WRITE (6,2002)
FZTM = DATA(12)
OEGD = DATA(7)
OMGSD = DATA(19)
FZWD = DATA(11)
FZASD = DATA(5)
WRITE (6,2006) OEGD,OMGSD,FZWD,FZASD,FZTM,FZTS,DIAZ,NSMP
2006 FORMAT (1.5X,*RESULTS FOR FOLLOWING CONDITIONS - *//,
C12X,*ITEM* 15X*MEAN*,4X,*STD DEV*//,
C10X,*ELEVATION*,4X,2F10.4,/,10X,*FUZE ANGLE*,3X,2F10.4,/,
C10X,*LINEAR FUZE*,2X,2F10.4,/,10X,*AZIMUTH *,F10.4,/
C10X,*SAMPLE SIZE - *15,/)
WRITE (6,2003) GMR,GMD,GMH
WRITE (6,2012)
2012 FORMAT (1.5X,*ERRR DATA*,17X,*PK*,3X,
C*PKFRAG PRADR PKCHIT PKBLST*)
2003 FORMAT (5X,*HOMING POINT COORDINATES (R,D,H) = *,
C 2(F6.1,*),F6.1)
DO 72 I=1,NLOOP
IF(NCEP.EQ.0) WRITE (6,2007) SDD(I),SDH(I),PKG(I)
C,PK(I),PKR(I),PKD(I),PKBL(I)
IF(NCEP.EQ.1) WRITE (6,2008) CEP(I),PKG(I)
C,PK(I),PKR(I),PKD(I),PKBL(I)
72 CONTINUE
WRITE (6,2002)
WRITE (6,1003)
DO 26 I=1,NLOOP
26 WRITE (6,1004) CEP(I),RRG(I),RSG(I),BRG(I),BDG(I),BDSG(I),
C ,BHG(I),BHSG(I)
55
915
920
925
930
935
940
945
950
955
960

```

```

PROGRAM ARP (INPUT=220,OUTPUT=220,TAPE5=INPUT,TAPE6=OUTPUT,
CTAPE1=220,TAPE2=220,TAPE3=220,TAPE4=220,TAPE8=220)
DIMENSION ARAY(5C),DATA(50),PK1(40,20,8)
DIMENSION PKS(12,6,10),PKR(50)
DIMENSION IKS(12,6,10),PKM(50)
DIMENSION HGT(9),XONG(3),INEW(50),VHT(5),GLTR(3,10)

```

```

000100
000110
000120
000130
000140
000150

```

```

73/74 OPT=1

```

```

PROGRAM ARP

```

```

1
PROGRAM ARP (INPUT=220,OUTPUT=220,TAPE5=INPUT,TAPE6=OUTPUT,
CTAPE1=220,TAPE2=220,TAPE3=220,TAPE4=220,TAPE8=220)
DIMENSION ARAY(5C),DATA(50),PK1(40,20,8)
DIMENSION PKS(12,6,10),PKR(50)
DIMENSION IKS(12,6,10),PKM(50)
DIMENSION HGT(9),XONG(3),INEW(50),VHT(5),GLTR(3,10)

```



```

      WRITE (6,*) * NOTE: FOLLOWING GUIDANCE ERROR PARAMETERS*
      WRITE (6,*) * (SIGD,SIGH) ARE MEASURED*
      WRITE (6,*) * IN PLANE NORMAL TO TRAJECTORY AND*
      WRITE (6,*) * PASSING THROUGH HOMING POINT*
      WRITE (6,*) * NGER - NUMBER OF GUIDANCE ERRORS TO CONSIDER*
      WRITE (6,*) * ENTER HOMING POINT (R,D,H), GUIDANCE*
      WRITE (6,*) * ERRORS ARE DISTRIBUTED ABOUT HOMING PT.*
      WRITE (6,*) * NCEP - 1., IF CEP IS INPUT FOR GUIDANCE ERROR SIGMAS* 000740
      WRITE (6,*) * FZAM,FZAS,FZTM,FZTS - FUZING ERROR OPTIONS* 000750
      WRITE (6,*) * FZAM - MEAN ANGLE AT WHICH FUZING OCCURS ON*
      WRITE (6,*) * INTERCEPT* 000760
      WRITE (6,*) * FZAS - STD DEV ASSOCIATED WITH FZAM* 000770
      WRITE (6,*) * NOTE: FUZE ANGLE IS CONSTRAINED TO (0,PI)* 000780
      WRITE (6,*) * NOTE: FOR UNIFORM FUZING ANGLE BETWEEN FZAM* 000790
      WRITE (6,*) * AND FZAS, ENTER A NEGATIVE VALUE FOR FZAM*
      WRITE (6,*) * FUZE ANGLE WILL BE CHOSEN UNIFORMLY RANDOM* 000800
      WRITE (6,*) * BETWEEN POSITIVE FZAM AND FZAS* 000810
      WRITE (6,*) * FOR TIME-TO-GO FUZE, ENTER NEGATIVE FZAS.* 000820
      WRITE (6,*) * FUZING PLANE PASSES THROUGH FUZING GLITTER*
      WRITE (6,*) * POINT NORMAL TO SAMPLE TRAJECTORY* 000830
      WRITE (6,*) * FZTM - MEAN DISTANCE FROM GUIDANCE PLANE AT WHICH* 000840
      WRITE (6,*) * FUZING WILL OCCUR ALONG TRAJECTORY* 000850
      WRITE (6,*) * FUZING WILL OCCUR FOR HEIGHT FUZING* 000860
      WRITE (6,*) * NOTE: ENTER A NEGATIVE ABS(FZTM)* 000870
      WRITE (6,*) * WITH MEAN HEIGHT ABS(FZTM)* 000880
      WRITE (6,*) * FZTS - STD DEV ASSOCIATED WITH FZTM* 000890
      WRITE (6,*) * SAMP - SAMPLE SIZE* 000900
      WRITE (6,*) * PKNH - NUMBER OF HEIGHTS AT WHICH FRAGMENTATION* 000910
      WRITE (6,*) * PK DATA WILL BE DEFINED* 000920
      WRITE (6,*) * NOTE: PKNH < 9* 000930
      WRITE (6,*) * PKPF - PROBABILITY OF PRIMARY FUZE FUNCTIONING* 000940
      WRITE (6,*) * PDVT - 0. FOR PG BACKUP, NVT FOR VT BACKUP FUZE* 000950
      WRITE (6,*) * WHERE NVT = NUMBER OF VT BURST HEIGHTS* 000960
      WRITE (6,*) * GLTR - 0. IF PRIMARY FUZE FUNCTIONS RELATIVE TO* 000970
      WRITE (6,*) * CENTER OF TARGET, NGLT IF PRIMARY FUZE* 000980
      WRITE (6,*) * FUNCTIONS RELATIVE TO ANY ONE OF NGLT* 000990
      WRITE (6,*) * SET NGLT NEGATIVE TO PICK FIRST* 001000
      WRITE (6,*) * POINT ENCOUNTERED.* 001010
      WRITE (6,*) * SRNG - MAXIMUM RANGE FOR COMPUTING PK VS RANGE* 001020
      WRITE (6,*) * PENT - 1. TO PRINT SUMMARY ONLY, 0. OTHERWISE* 001030
      WRITE (6,*) * DEBUG - 6. TO PRINTOUT PROGRAM DEBUGGING DATA* 001040
      WRITE (6,*) * DEBUG = 1, GUIDANCE & FUZING DATA* 001050
      WRITE (6,*) * DEBUG = 2, DIRECT HIT PENETRATION DATA* 001060
      WRITE (6,*) * DEBUG = 4, PK BOX DATA* 001070
      WRITE (6,*) * DEBUG = 5, PK GRIDS* 001080
      WRITE (6,*) * DEBUG = 6, PK VS R DATA* 001090
      WRITE (6,*) * TGTIC - HEIGHT OF TARGET CENTER ABOVE GROUND* 001100
      WRITE (6,*) * QUDR - DUD RATE OF PROJECTILE, EXPRESSED AS A FRACTION* 001110
      CN*          WRITE (6,*) * CHIT - DIRECT HIT OPTION, NUMBER OF TARGET BOXES* 001120
      WRITE (6,*) * IF DHIT IS OMITTED AND BLST IS INCLUDED,* 001130
      WRITE (6,*) * BLST IS RADIUS FROM (0,0,TGTC) WITHIN* 001140
      WRITE (6,*) * WHICH PKBLST = 1.* 001150
      WRITE (6,*) * PKDH - DIRECT HIT PK (0. = 1.)* 001160
      WRITE (6,*) * PKBL - BLAST PK (0. = 1.)* 001170
      WRITE (6,*) * PRADR - 1., DEFINE FUNC FOR BLAST KILL OF RADAR ONLY* 001180
      WRITE (6,*) * AND READ IN RADAR ANTENNA COORDINATES.* 001190
      WRITE (6,*) * 001200
      WRITE (6,*) * 001210
      WRITE (6,*) * 001220
      WRITE (6,*) * 001230
  
```

```

115      WRITE (6,*) "TO DEFINE FUNC, SPECIFY R1 AND R2."
          WRITE (6,*) "WHERE BLAST PK IS 1 OUT TO R1 AND"
          WRITE (6,*) "DECLINES LINEARLY TO 0 AT R2."
          WRITE (6,*) "DHAZ - AZIMUTH ANGLE OF ATTACK OFF FRONT OF TARGET"
          WRITE (6,*) "TOWARD DRIVER SIDE. SET TO -1. FOR RANDOM"
          WRITE (6,*) "BLAST RADIUS WITHIN WHICH VEHICLE PK=PKBL."
          WRITE (6,*) "BLST - BURST HEIGHT"
          WRITE (6,*) "NOTE: ENTER BLAST RADII VS. BURST HEIGHT"
          WRITE (6,*) "ENTER NEGATIVE NUMBER OF BLST, MGT PAIRS"
          WRITE (6,*) "IN PLACE OF VALUE OF BLST. PAIRS OF"
          WRITE (6,*) "BLST,MGT ARE ENTERED IN ASCENDING ORDER"
          WRITE (6,*) "OF HEIGHT."
          WRITE (6,*) "COORDINATE SYSTEM IS RECTANGULAR."
          WRITE (6,*) "TARGET HEADING IS NEGATIVE RANGE."
          WRITE (6,*) "DRIVER SIDE (LEFT) IS POSITIVE DEFLECTION."
          WRITE (6,*) "HEIGHT IS MEASURED FROM GROUND."
54      NPRT = 0
          ISET = 0
          ITIME = 0
          CALL RDOUT (INIT)
15      CALL RDIN (INIT)
          ISET = 1
          IF (IRD.EQ.5) GO TO 88
          IF (NPRT.GT.0) GO TO 80
          WRITE (6,*) "ENTER DATA BY ENTERING CODE NAME"
          WRITE (6,*) "FOLLOWED BY A COMMA AND THE VALUE IN FLOATING"
          WRITE (6,*) "POINT FORMAT. TO END DATA ENTRY, ENTER"
          WRITE (6,*) "THE WORD END IN COLUMNS 1-3."
C       FILE TAPE1 CONTAINS BASIC INPUT DATA
C       FILES TAPE2 - TAPE4 CONTAIN FRAGMENTATION PK GRIDS
C       FOR DIFFERENT ANGLES OF ATTACK
135      C
          BS WRITE (6,*) "DO YOU WISH TO INITIALIZE DATA FROM"
          WRITE (6,*) "DATA FILE TAPE1?"
          READ (5,1001) ANS
          IRD = 5
          IF (ANS.EQ.YES) IRD = 1
          80 REWIND 1
          REWIND 2
          REWIND 3
          REWIND 4
          PI = ATAN2(0.,-1.)
          DO 51 I=1,10
          51 PKG(I) = 0.
C       INITIALIZE OR UPDATE DATA
140      C
          REWIND 1
          7 IF (IRD.EQ.5) WRITE (6,*) "ENTER DATA OR END - "
          READ (IRD,100) AAAA,VALUE
          1000 FORMAT (A4,1X,F10.3)
          IF (AAEA.EQ.END) GO TO 14
          DO 53 J=1,50
          IF (AAAE.NE.AAN(J)) GO TO 53
          INEX(J) = 1
          DATA(J) = VALUE
          52 TU 7
58      145
          150
          155
          160
          165
          170
          175
          180
          185
          190
          195
          200
          205
          210
          215
          220
          225
          230
          235
          240
          245
          250
          255
          260
          265
          270
          275
          280
          285
          290
          295
          300
          305
          310
          315
          320
          325
          330
          335
          340
          345
          350
          355
          360
          365
          370
          375
          380
          385
          390
          395
          400
          405
          410
          415
          420
          425
          430
          435
          440
          445
          450
          455
          460
          465
          470
          475
          480
          485
          490
          495
          500
          505
          510
          515
          520
          525
          530
          535
          540
          545
          550
          555
          560
          565
          570
          575
          580
          585
          590
          595
          600
          605
          610
          615
          620
          625
          630
          635
          640
          645
          650
          655
          660
          665
          670
          675
          680
          685
          690
          695
          700
          705
          710
          715
          720
          725
          730
          735
          740
          745
          750
          755
          760
          765
          770
          775
          780
          785
          790
          795
          800
          805
          810
          815
          820
          825
          830
          835
          840
          845
          850
          855
          860
          865
          870
          875
          880
          885
          890
          895
          900
          905
          910
          915
          920
          925
          930
          935
          940
          945
          950
          955
          960
          965
          970
          975
          980
          985
          990
          995
          1000
          1005
          1010
          1015
          1020
          1025
          1030
          1035
          1040
          1045
          1050
          1055
          1060
          1065
          1070
          1075
          1080
          1085
          1090
          1095
          1100
          1105
          1110
          1115
          1120
          1125
          1130
          1135
          1140
          1145
          1150
          1155
          1160
          1165
          1170
          1175
          1180
          1185
          1190
          1195
          1200
          1205
          1210
          1215
          1220
          1225
          1230
          1235
          1240
          1245
          1250
          1255
          1260
          1265
          1270
          1275
          1280
          1285
          1290
          1295
          1300
          1305
          1310
          1315
          1320
          1325
          1330
          1335
          1340
          1345
          1350
          1355
          1360
          1365
          1370
          1375
          1380
          1385
          1390
          1395
          1400
          1405
          1410
          1415
          1420
          1425
          1430
          1435
          1440
          1445
          1450
          1455
          1460
          1465
          1470
          1475
          1480
          1485
          1490
          1495
          1500
          1505
          1510
          1515
          1520
          1525
          1530
          1535
          1540
          1545
          1550
          1555
          1560
          1565
          1570
          1575
          1580
          1585
          1590
          1595
          1600
          1605
          1610
          1615
          1620
          1625
          1630
          1635
          1640
          1645
          1650
          1655
          1660
          1665
          1670
          1675
          1680
          1685
          1690
          1695
          1700
          1705
          1710
          1715
          1720
          1725
          1730
          1735
          1740
          1745
          1750
          1755
          1760
          1765
          1770
          1775
          1780
          1785
          1790
          1795
          1800
          1805
          1810
          1815
          1820
          1825
          1830
          1835
          1840
          1845
          1850
          1855
          1860
          1865
          1870
          1875
          1880
          1885
          1890
          1895
          1900
          1905
          1910
          1915
          1920
          1925
          1930
          1935
          1940
          1945
          1950
          1955
          1960
          1965
          1970
          1975
          1980
          1985
          1990
          1995
          2000
          2005
          2010
          2015
          2020
          2025
          2030
          2035
          2040
          2045
          2050
          2055
          2060
          2065
          2070
          2075
          2080
          2085
          2090
          2095
          2100
          2105
          2110
          2115
          2120
          2125
          2130
          2135
          2140
          2145
          2150
          2155
          2160
          2165
          2170
          2175
          2180
          2185
          2190
          2195
          2200
          2205
          2210
          2215
          2220
          2225
          2230
          2235
          2240
          2245
          2250
          2255
          2260
          2265
          2270
          2275
          2280
          2285
          2290
          2295
          2300
          2305
          2310
          2315
          2320
          2325
          2330
          2335
          2340
          2345
          2350
          2355
          2360
          2365
          2370
          2375
          2380
          2385
          2390
          2395
          2400
          2405
          2410
          2415
          2420
          2425
          2430
          2435
          2440
          2445
          2450
          2455
          2460
          2465
          2470
          2475
          2480
          2485
          2490
          2495
          2500
          2505
          2510
          2515
          2520
          2525
          2530
          2535
          2540
          2545
          2550
          2555
          2560
          2565
          2570
          2575
          2580
          2585
          2590
          2595
          2600
          2605
          2610
          2615
          2620
          2625
          2630
          2635
          2640
          2645
          2650
          2655
          2660
          2665
          2670
          2675
          2680
          2685
          2690
          2695
          2700
          2705
          2710
          2715
          2720
          2725
          2730
          2735
          2740
          2745
          2750
          2755
          2760
          2765
          2770
          2775
          2780
          2785
          2790
          2795
          2800
          2805
          2810
          2815
          2820
          2825
          2830
          2835
          2840
          2845
          2850
          2855
          2860
          2865
          2870
          2875
          2880
          2885
          2890
          2895
          2900
          2905
          2910
          2915
          2920
          2925
          2930
          2935
          2940
          2945
          2950
          2955
          2960
          2965
          2970
          2975
          2980
          2985
          2990
          2995
          3000
          3005
          3010
          3015
          3020
          3025
          3030
          3035
          3040
          3045
          3050
          3055
          3060
          3065
          3070
          3075
          3080
          3085
          3090
          3095
          3100
          3105
          3110
          3115
          3120
          3125
          3130
          3135
          3140
          3145
          3150
          3155
          3160
          3165
          3170
          3175
          3180
          3185
          3190
          3195
          3200
          3205
          3210
          3215
          3220
          3225
          3230
          3235
          3240
          3245
          3250
          3255
          3260
          3265
          3270
          3275
          3280
          3285
          3290
          3295
          3300
          3305
          3310
          3315
          3320
          3325
          3330
          3335
          3340
          3345
          3350
          3355
          3360
          3365
          3370
          3375
          3380
          3385
          3390
          3395
          3400
          3405
          3410
          3415
          3420
          3425
          3430
          3435
          3440
          3445
          3450
          3455
          3460
          3465
          3470
          3475
          3480
          3485
          3490
          3495
          3500
          3505
          3510
          3515
          3520
          3525
          3530
          3535
          3540
          3545
          3550
          3555
          3560
          3565
          3570
          3575
          3580
          3585
          3590
          3595
          3600
          3605
          3610
          3615
          3620
          3625
          3630
          3635
          3640
          3645
          3650
          3655
          3660
          3665
          3670
          3675
          3680
          3685
          3690
          3695
          3700
          3705
          3710
          3715
          3720
          3725
          3730
          3735
          3740
          3745
          3750
          3755
          3760
          3765
          3770
          3775
          3780
          3785
          3790
          3795
          3800
          3805
          3810
          3815
          3820
          3825
          3830
          3835
          3840
          3845
          3850
          3855
          3860
          3865
          3870
          3875
          3880
          3885
          3890
          3895
          3900
          3905
          3910
          3915
          3920
          3925
          3930
          3935
          3940
          3945
          3950
          3955
          3960
          3965
          3970
          3975
          3980
          3985
          3990
          3995
          4000
          4005
          4010
          4015
          4020
          4025
          4030
          4035
          4040
          4045
          4050
          4055
          4060
          4065
          4070
          4075
          4080
          4085
          4090
          4095
          4100
          4105
          4110
          4115
          4120
          4125
          4130
          4135
          4140
          4145
          4150
          4155
          4160
          4165
          4170
          4175
          4180
          4185
          4190
          4195
          4200
          4205
          4210
          4215
          4220
          4225
          4230
          4235
          4240
          4245
          4250
          4255
          4260
          4265
          4270
          4275
          4280
          4285
          4290
          4295
          4300
          4305
          4310
          4315
          4320
          4325
          4330
          4335
          4340
          4345
          4350
          4355
          4360
          4365
          4370
          4375
          4380
          4385
          4390
          4395
          4400
          4405
          4410
          4415
          4420
          4425
          4430
          4435
          4440
          4445
          4450
          4455
          4460
          4465
          4470
          4475
          4480
          4485
          4490
          4495
          4500
          4505
          4510
          4515
          4520
          4525
          4530
          4535
          4540
          4545
          4550
          4555
          4560
          4565
          4570
          4575
          4580
          4585
          4590
          4595
          4600
          4605
          4610
          4615
          4620
          4625
          4630
          4635
          4640
          4645
          4650
          4655
          4660
          4665
          4670
          4675
          4680
          4685
          4690
          4695
          4700
          4705
          4710
          4715
          4720
          4725
          4730
          4735
          4740
          4745
          4750
          4755
          4760
          4765
          4770
          4775
          4780
          4785
          4790
          4795
          4800
          4805
          4810
          4815
          4820
          4825
          4830
          4835
          4840
          4845
          4850
          4855
          4860
          4865
          4870
          4875
          4880
          4885
          4890
          4895
          4900
          4905
          4910
          4915
          4920
          4925
          4930
          4935
          4940
          4945
          4950
          4955
          4960
          4965
          4970
          4975
          4980
          4985
          4990
          4995
          5000
          5005
          5010
          5015
          5020
          5025
          5030
          5035
          5040
          5045
          5050
          5055
          5060
          5065
          5070
          5075
          5080
          5085
          5090
          5095
          5100
          5105
          5110
          5115
          5120
          5125
          5130
          5135
          5140
          5145
          5150
          5155
          5160
          5165
          5170
          5175
          5180
          5185
          5190
          5195
          5200
          5205
          5210
          5215
          5220
          5225
          5230
          5235
          5240
          5245
          5250
          5255
          5260
          5265
          5270
          5275
          5280
          5285
          5290
          5295
          5300
          5305
          5310
          5315
          5320
          5325
          5330
          5335
          5340
          5345
          5350
          5355
          5360
          5365
          5370
          5375
          5380
          5385
          5390
          5395
          5400
          5405
          5410
          5415
          5420
          5425
          5430
          5435
          5440
          5445
          5450
          5455
          5460
          5465
          5470
          5475
          5480
          5485
          5490
          5495
          5500
          5505
          5510
          5515
          5520
          5525
          5530
          5535
          5540
          5545
          5550
          5555
          5560
          5565
          5570
          5575
          5580
          5585
          5590
          5595
          5600
          5605
          5610
          5615
          5620
          5625
          5630
          5635
          5640
          5645
          5650
          5655
          5660
          5665
          5670
          5675
          5680
          5685
          5690
          5695
          5700
          5705
          5710
          5715
          5720
          5725
          5730
          5735
          5740
          5745
          5750
          5755
          5760
          5765
          5770
          5775
          5780
          5785
          5790
          5795
          5800
          5805
          5810
          5815
          5820
          5825
          5830
          5835
          5840
          5845
          5850
          5855
          5860
          5865
          5870
          5875
          5880
          5885
          5890
          5895
          5900
          5905
          5910
          5915
          5920
          5925
          5930
          5935
          5940
          5945
          5950
          5955
          5960
          5965
          5970
          5975
          5980
          5985
          5990
          5995
          6000
          6005
          6010
          6015
          6020
          6025
          6030
          6035
          6040
          6045
          6050
          6055
          6060
          6065
          6070
          6075
          6080
          6085
          6090
          6095
          6100
          6105
          6110
          6115
          6120
          6125
          6130
          6135
          6140
          6145
          6150
          6155
          6160
          6165
          6170
          6175
          6180
          6185
          6190
          6195
          6200
          6205
          6210
          6215
          6220
          6225
          6230
          6235
          6240
          6245
          6250
          6255
          6260
          6265
          6270
          6275
          6280
          6285
          6290
          6295
          6300
          6305
          6310
          6315
          6320
          6325
          6330
          6335
          6340
          6345
          6350
          6355
          6360
          6365
          6370
          6375
          6380
          6385
          6390
          6395
          6400
          6405
          6410
          6415
          6420
          6425
          6430
          6435
          6440
          6445
          6450
          6455
          6460
          6465
          6470
          6475
          6480
          6485
          6490
          6495
          6500
          6505
          6510
          6515
          6520
          6525
          6530
          6535
          6540
          6545
          6550
          6555
          6560
          6565
          6570
          6575
          6580
          6585
          6590
          6595
          6600
          6605
          6610
          6615
          6620
          6625
          6630
          6635
          6640
          6645
          6650
          6655
          6660
          6665
          6670
          6675
          6680
          6685
          6690
          6695
          6700
          6705
          6710
          6715
          6720
          6725
          6730
          6735
          6740
          6745
          6750
          6755
          6760
          6765
          6770
          6775
          6780
          6785
          6790
          6795
          6800
          6805
          6810
          6815
          6820
          6825
          6830
          6835
          6840
          6845
          6850
          6855
          6860
          6865
          6870
          6875
          6880
          6885
          6890
          6895
          6900
          6905
          6910
          6915
          6920
          6925
          6930
          6935
          6940
          6945
          6950
          6955
          6960
          6965
          6970
          6975
          6980
          6985
          6990
          6995
          7000
          7005
          7010
          7015
          7020
          7025
          7030
          7035
          7040
          7045
          7050
          7055
          7060
          7065
          7070
          7075
          7080
          7085
          7090
          7095
          7100
          7105
          7110
          7115
          7120
          7125
          7130
          7135
          7140
          7145
          7150
          7155
          7160
          7165
          7170
          7175
          7180
          7185
          7190
          7195
          7200
          7205
          7210
          7215
          7220
          7225
          7230
          7235
          7240
          7245
          7250
          7255
          7260
          7265
          7270
          7275
          7280
          7285
          7290
          7295
          7300
          7305
          7310
          7315
          7320
          7325
          7330
          7335
          7340
          7345
          7350
          7355
          7360
          7365
          7370
          7375
          7380
          7385
          7390
          73
```

PROGRAM AAA
 73/74 OPT=1
 FTN 4,8+569
 03/13/61 66,26,26
 PAGE 4

```

53 CONTINUE
      WRITE (5,2000) AAAA
      GO TO 7
175   14 CALL READ (DATA,INEX,ANAM,IRD,1,ROH,DDH,HDH)
      C
      SET UP TAPE
      C
      9 REWIND 1
      DO 61 I=1,50
      IF(DATA(I,EQ.0.)) GO TO 8
      WRITE (1,1000) ANAM(I),DATA(I)
      CONTINUE
      WRITE (1,1000) END
      CALL WRITE (DATA,I,CEP,RESH,DDH,HDH)
      REWIND 1
      15 ITIME = EC(5) GO TO 12
      WRITE (5,1) "DO YOU WANT CURRENT INPUT LISTED?"
      READ (5,1) A,B
      IF(ANS,NE,YES) GO TO 23
      IF(ITIME.GT.0) WRITE (5,*) "CURRENT DATA - "
      12 IF(ITIME.EQ.0) WRITE (5,*) "INITIAL INPUTS - "
      C
      LIST DATA FILE (TAPE)
      C
      DO 6 1=1,50
      READ (1,1000) A,B
      IF(A,EO,END) GO TO 6
      8 WRITE (6,1002) A,B
      1002 FORMAT (1X,A4,1X,F10.3)
      6 WRITE (6,1C,2) END
      CALL WRITE (DATA,6,CEP,ROH,DDH,HDH)
      23 REWIND 1
      ITIME = ITIME + 1
      IF(ISET,EQ,1) GO TO 86
      WRITE (6,*) "DO YOU WANT TO CHANGE ANY DATA? - "
      READ (5,1001) ANS
      IF(ANS,NE,YES) GO TO 82
      89 ISET = 0
      C
      READ IN CHANGES
      C
      DO 13 I=1,50
13     INEX(I) = 0
      DO 2 I=1,1000
      WRITE (6,*) "ENTER DATA OR END - "
      READ (5,1007) AAAA,VALUE
      IF(AAAA,EQ,END) GO TO 3
      1001 FORMAT (A11)
      DO 4 J=1,50
      IF(AAAA,NE,ANAM(J)) GO TO 4
      DATA(J) = VALUE
      INEX(J) = 1
      GO TO 2
      4 CONTINUE
      WRITE (6,2000) AAAA
      2000 FORMAT (1X,"***** DO NOT RECOGNIZE ",A," *****")
      2 CONTINUE
  
```

```

3 CALL READ (DATA,INEW,ANAM,S,O,RDH,D3H,HDH)      652350
   GO TO 9                                              652350
   82 DO 83 I=1,50                                     652350
   83 INEW(I) = 0                                       652410
   C
   C     SET UP DATA
   C
235   C     LOAD INPUT DATA INTO VARIABLE SET
   C     AND CONVERT DEGREES TO RADIANS
   C
   FZAM = DATA(1)/57.29578                           652420
   FZIM = ABS(DATA(2))                                652420
   PKDHX = DATA(3)                                    652420
   PKBLX = DATA(4)                                    652420
   FZAS = DATA(5) = -7.29578                         652420
   ITTG = 0                                           652420
   IF(FZAS.LT.-0.) ITTG = 1                          652420
   FZAS = ASG(FZAS)                                 652420
   FZTS = DATA(6)                                    652420
   OMEG = DATA(7)/57.29578                         652420
   NGER = DATA(8)                                    652420
   NCEP = DATA(9)                                    652420
   IFUN = 0                                           652420
   NDHT = DATA(11)                                  652420
   NSHP = DATA(12)                                  652420
   NRDR = DATA(13)                                  652420
   DHAZ = DATA(14)/57.29578                         652420
   NH = DATA(17)                                    652420
   NA = 0                                           652420
   CMGS = 'C.'                                     652420
   PKPF = DATA(21)                                  652420
   NVT = DATA(20)                                   652420
   NGLT = DATA(22)                                  652420
   JGLT = 1                                         652420
   JGLT = ISIGN(JGLT,NGLT)                         652420
   NGLT = IABS(NGLT)                               652420
   SNGC = DATA(23)                                  652420
   NPRT = DATA(24)                                  652420
   NDSG = DATA(25)                                  652420
   TGIC = DATA(26)                                  652420
   DUOR = DATA(27)                                  652420
   BLST = DATA(28)                                  652420
   IF(BLST.LE.0.) GO TO 94                         652420
   BLST(1) = BLST
   MBLST(1) = 106000.
   BLST = 1                                         652420
   94 NBLST = ABS(BLST)                            652420
   IHFZ = 0                                         652420
   IF(DATA(2).LT.0.) IHFZ = 1                      652420
   IF(PKDHX.EQ.0.) PKDHX = 1.                      652420
   IF(PKBLX.EQ.0.) PKBLX = 1.                      652420
   NLCP = NGER                                     652420
   IF(NLCP.EQ.1) WRITE (6,* ) "DEBUG OPTION ",NDEG 652420
   IF(DATA(2).NE.0.) IFUZ = 2                      652420
   IF(DATA(1).NE.0.) IFUZ = 1                      652420
   XNG = C                                         652420
   IF(XNG.EQ.0) GO TO 115                         652420

```

```

6          PAGE      08.73.20
PAGE      03/13/81   FTR 4.8+508

PROGRAM ARP    73/74   CFT-1

DO 116 I=1,NDHT
  RRNG = 10000.
  IF(SIGN(1.,DDH(I,1)).EQ.SIGN(1.,DDH(I,2))) GO TO 118
  IF(SIGN(1.,RDH(I,1)).EQ.SIGN(1.,RDH(I,2))) GO TO 115
  DO 119 J=1,2
    RRNG = AMIN1(RRNG,DDH(I,J))
    RRNG = AMIN1(RRNG,RDH(I,J))
  119  RRNG = SQRT(RRNC**2.+RDH(I,1)**2.)
  GO TO 116

116  CONTINUE
  XRNG = RDH(I,1)
  XRNG = AMIN1(XRNG,RDH(NDHT,2))

118  XRNG = RDH(I,1)
  116  CONTINUE
  XRNG = AMIN1(XRNG,RDH(NDHT,2))

115  IF(SRNG.EQ.0.) SRNG = 100.
  DL = ALOG(SRNG-XRNG)/10.
  DO 111 I=1,10
    XI = I
    111  RANGE(I) = XRNG + EXP(DL*X1)
    RANGE(I) = 1000.
    IF(NINT.LE.-1) GO 10 67
    DC 68 I=2,NVT
    6B  PVT(I) = PVT(I) + PVT(I-1)
    67  IF(NGLT.GT.0) GO TG 59
    DO 60 I=1,3
    60  GLTR(I,1) = 0.
    59  IF(NA.EQ.0) GO 10 48
    DO 28 I=1,NA
    28  XCNG(I) = XGN(GI)/57.29578
    4B  CONTINUE
C     READ IN PK GRIDS FOR EACH ATTACK ANGLE/BURST HEIGHT
C     COMBINATION
C     IF(NH.EQ.0) GO TO 78
C     CALL GRIDS (PK1,NH,2,RGRD,DGRD,MR,ND,MDBG)
C     LOOP OVER SIMULATIONS FOR EACH GUIDANCE ERROR SET
C     78 DO 69 ILUP=1,NLCOP
C     C     INITIALIZE COUNTERS
C     DO 70 I=1,50
C       PKM(I) = 0.
C     70  IKM(I) = 0.
C     DO 52 I=1,12
C     DO 52 J=1,6
C     DO 52 K=1,10
C       IK5(I,J,K) = 0
C     52  PK5(I,J,K) = 0.
C       PKRADR = 0.
C       PKDHIT = 0.
C       PKBASE = 0.
C       PKBLT = 0.
C       PKTOT = 0.
C       PKTOT2 = 0.
C       PKBAR = 0.
C       RREAR2 = 0.
C     325  C
C     330  C
C     335  C
C     340  C

```

PROGRAM APP 73/74 OPT=1 FTN 4.8+508 03/13/81 08.29.30 PAGE 7
 345 C BOBAR = 0.
 BDBAR2 = 0.
 BRBAR = 0.
 BRBAR2 = 0.
 BHBAR = 0.
 BHBAR2 = 0.
 IF(PKPF.EQ.0.) PKPF = 1.
 IF(PKPF.LT.0.) PKPF = 0.
 SIGD = SDD(1LLP)
 SIGH = SDH(1LP)
 NCT = 0.
 350 C BEGIN SIMULATIONS
 C D3 1 ISIM=1,NSMP
 IF(DATA(16).LT.0.) DHAZ = RDM(1)*2.*PI
 PKAMP = 0.0
 PKDH = 0.
 PKBLST = 0.
 PKRDR = 0.
 355 C CHECK FOR END
 IF(RDM(1).LE.DUDR) GO TO 16
 C SAMPLE FRM ATTACK ANGLE DISTRIBUTION
 C CALL BOXNO (Z1,Z2)
 OMEGA = Z1*CMGS + ONEG
 SINO = SIN(OMEGA)
 COSO = COS(OMEGA)
 TANO = 1.
 IF(COSD.NE.0.) TANO = SINO/COSO
 360 C ROTATE COORDINATES OF HOMING POINT ACCORDING
 TO AZIMUTH COMPONENT OF ATTACK ANGLE.
 365 C ALL COMPUTATIONS TO DETERMINE FUZING POINT ARE IN
 ROTATED COORDINATE SYSTEM.
 C
 370 C
 62
 375 C
 380 C
 385 C
 390 C
 395 C

BOBAR = 0.
 BDBAR2 = 0.
 BRBAR = 0.
 BRBAR2 = 0.
 BHBAR = 0.
 BHBAR2 = 0.
 IF(PKPF.EQ.0.) PKPF = 1.
 IF(PKPF.LT.0.) PKPF = 0.
 SIGD = SDD(1LLP)
 SIGH = SDH(1LP)
 NCT = 0.
 350 C BEGIN SIMULATIONS
 C D3 1 ISIM=1,NSMP
 IF(DATA(16).LT.0.) DHAZ = RDM(1)*2.*PI
 PKAMP = 0.0
 PKDH = 0.
 PKBLST = 0.
 PKRDR = 0.
 355 C CHECK FOR END
 IF(RDM(1).LE.DUDR) GO TO 16
 C SAMPLE FRM ATTACK ANGLE DISTRIBUTION
 C CALL BOXNO (Z1,Z2)
 OMEGA = Z1*CMGS + ONEG
 SINO = SIN(OMEGA)
 COSO = COS(OMEGA)
 TANO = 1.
 IF(COSD.NE.0.) TANO = SINO/COSO
 360 C ROTATE COORDINATES OF HOMING POINT ACCORDING
 TO AZIMUTH COMPONENT OF ATTACK ANGLE.
 365 C ALL COMPUTATIONS TO DETERMINE FUZING POINT ARE IN
 ROTATED COORDINATE SYSTEM.
 C
 370 C
 62
 375 C
 380 C
 385 C
 390 C
 395 C

003520
 003530
 003540
 003550
 003560
 003570
 003580
 003590
 003600
 003610
 003620
 003630
 003640
 003650
 003660
 003670
 003680
 003690
 003700
 003710
 003720
 003730
 003740
 003750
 003760
 003770
 003780
 003790
 003800
 003810
 003820
 003830
 003840
 003850
 003860
 003870
 003880
 003890
 003900
 003910
 003920
 003930
 003940
 003950
 003960
 003970
 003980
 003990
 004000
 004010
 004020
 004030
 004040
 004050
 004060
 004070
 004080

GMRR = GMRR
 GMCR = GMCR
 CALL ROTATE (CMERR,GMDR,DHAZ,1.)
 SAMPLE FRM GUIDANCE ERROR DISTRIBUTION
 RELATIVE TO HOMING POINT
 CALL BOXNO (D,H)
 DMIN = SQRT((SIGH**H)**2. + (SIGD*D)**2.)
 GR = GMRR + SIGP*H*SING
 GD = GMCR + SIGD*D
 GH = GMH + SIGH*H*CCSC
 (GR,GD,GH) IS INTERCEPT OF
 TRAJECTORY WITH GUIDANCE PLANE
 (RF,DF,HF) WILL BE FUZING POINT ON TRAJECTORY.

R

	PROGRAM ARP	73/74 OPT=1	FTN 4.8+508	03/13/81	08.25.30	PAGE
400	C RF = GR DF = GD HF = GH			004090 004100 004110 004120 004130 004140 004150 004160 004170 004180 004190 004200 004210 004220 004230 004240 004250 004260 004270 004280 004290 004300 004310 004320 004330 004340 004350 004360 004370 004380 004390 004400 004410 004420 004430 004440 004450 004460 004470 004480 004490 004500 004510 004520 004530 004540 004550 004560 004570 004580 004590 004600 004610 004620 004630 004640 004650		
405	C CHECK FOR PRIMARY FUZE FUNCTION					
	I8KUP = 0					
	IF(RDM(1).GT.PKPF) GO TO 16					
410	C CHECK FOR HEIGHT FUZING					
	IF(IFHZ.EQ.1) GO TO 74					
	Q2 = 0.					
415	C CHECK FOR APPROPRIATE FUZING					
	CALL BOXNO (21,22)					
	IGO = IFU2 + 1					
	IF(NSBG.GE.1) WRITE (6,5003) IFUZ,IGO,GR,GO,GH					
	GO TO (85,75,22,85),IGO					
420	C CHOOSE GLITTER POINT FOR FUZING. ANGULAR FUZE ONLY					
	75 IF(GLT.LT.0.AND.NGLT.GT.1) GO TO 76					
	XGLT = NGLT					
	IGLT = 1RDW(1)-0.0001*XGLT + 1.0					
	IF(IGLT.EQ.0) IGLT = 1					
	B6 RGLT = GLTR(1,IGLT)					
	DGLT = GLTR(2,IGLT)					
	HGLT = GLTR(3,IGLT)					
	IGC = 1					
	GO TO 77					
	76 ICG = NGLT					
	GRMAX = -100000.					
	77 DC E: IG_L=1,IGC					
	IF((DC, EQ. 1) GO TO 21					
	RGLT = GLTR(1,IGL)					
	DGLT = GLTR(2,IGL)					
	HGLT = GLTR(3,IGL)					
425	C ROTATE GLITTER POINT INTO ARP COORDINATE SYSTEM					
	21 IF(NDBG.EQ.1) WRITE (6,*,"RGLT,DGLT,HGLT = ",RGLT,DGLT,HGLT					
	CALL ROTATE (RGLT,DGLT,DHAZ2,1.)					
	IF(NDBG.EQ.1) WRITE (6,*,"ROTATED GLITTER POINT = "					
	IF(NDBG.EQ.1) WRITE (6,*,"DHAZ,RGLT,BGLT = ",DHAZ,RGLT,BGLT,004540					
	C HGLT					
	5003 FORMAT (1X,*IFUZ,IGO = *,2(I2,*,*,1X),*GR,GO,GH = *,3(F6.1,*,*,1X))004570					
430	C					
435	C USE LAW OF SINES AND LAW OF COSINES TO FIND					
	FUZING POINT ON TRAJECTORY. FIRST PICK A POINT					
	ALONG TRAJECTORY TO COMPUTE BETAK (ANGLE BETWEEN					
	TRAJECTORY AND A LINE (AB) FROM GLITTER POINT					
	(RGLT,CGLT,HGLT) TO Guidance PLANE INTERCEPT					
	(GR,GO,GH) - NOTE THAT EVERYTHING IS IN ROTATED					
440	C					
445	C					
450	C					
455	C					

PROGRAM APP	73/74	GPT*1	FTN 4.8+508	03/13/81	08.29.30	PAGE
						9
460	C	COORDINATE SYSTEM (THROUGH AZIMUTH ATTACK ANGLE COMPONENT). THEN, KNOWING BETAX AND FUZING ANGLE (ANG) COMPUTE ANGLE (GAMMA) WITH ITS VERTEX AT GLITTER POINT AND OPPOSITE TRAJECTORY SEGMENT BOUNDED BY GUIDANCE PLANE INTERCEPT AND FUZING POINT. FINALLY, KNOWING GAMMA, AB, AND ANG, COMPUTE O2, THE DISTANCE FROM GUIDANCE PLANE INTERCEPT TO FUZING POINT (USING THE LAW OF SINES).	004660 004670 004680 004690 004700 004710 004720 004730 004740 004750 004760 004770 004780 004790 004800 004810 004820 004830 004840 004850 004860 004870 004880 004890 004900 004910 004920 004930 004940 004950 004960 004970 004980 004990 005000 005010 005020 005030 005040 005050 005060 005070 005080 005090 005100 005110 005120 005130 005140 005150 005160 005170 005180 005190 005200 005210 005220			
465	C	TANGX = TANG IF(SING.EQ.0.) TANOX = 1. CB = 10. IF(SING.NE.0.) CB = CB/SINC				
470	C	GRL,GDL,GHL ARE COORDINATES OF A POINT ON THE TRAJECTORY USED TO COMPUTE BETAX.				
475	C	GRL = GR - 10./TANOX GDL = GD GHL = GH IF(SINO.NE.0.) GHL = GH + 10. AB2 = (RGLT-GR)**2. + (DGLT-GD)**2. + (HGLT-GH)**2. BB2 = (RGLT-GRL)**2. + (DGLT-GDL)**2. + (HGLT-GHL)**2. AB = SQRT(AB2)				
480	C	USE LAW OF COSINES TO COMPUTE BETAX, ANGLE WITH VERTEX AT GLITTER POINT AND OPPOSITE TRAJECTORY SEGMENT BOUNDED BY GUIDANCE PLANE INTERCEPT AND FUZING POINT.				
485	C	BETAX = ACOS((AB2-BB2+CB*CB)/(2.*AB*CB)) IF(NDBG.EQ.1) WRITE(6,*), "BETAX,GRL,GDL,GHL,AB,CB = ", C BETAX,GRL,GDL,GHL,AB,CB FZASX = FZAS IF(ITTG.EQ.1) FZASX = 0.				
490	C	ANGULAR FUZING FUNCTION				
495	C	ANG = Z2*FZASX + FZAM IF(FZAM.LT.0.) ANG = FZAM + RDM(1)*(FZASX-FZAM) IF(ANG.LT.-0.1745) GO TO 18 IF(ANG.GT.PI) GO TO 16				
500	C	O2 IS DISTANCE ALONG TRAJECTORY FROM GUIDANCE PLANE INTERCEPT TO FUZING POINT.				
505	C	GAMMA = PI - EETAX - ANG C IF GAMMA.LT.ZERO, USE SUPPLEMENT OF ANG FOR FUZING. C IF(GAMMA.LT.0.) ANG = PI - ANG O2 = AB*(SIN(GAMMA)/SIN(ANG)) IF(NDBG.EQ.1) WRITE(6,*), "O2,GAMMA,ANG = ",O2,GAMMA,ANG IF(DO.EQ.1) GO TO 22 IF(O2.LT.GRMAX) GO TO 84 GRMAX = O2 IGLT = 1GL 84 CONINUE				

PROGRAM ARP	73/74	OPT=1	FTN 4.8+SCS	03/13/81	08.29.30	PAGE 10
515	C	GO TO 66				
	C	LINEAR FUZING FUNCTION (ALONG TRAJECTORY)				
	C	FUZING DIRECTION IS POSITIVE IN THE NEGATIVE				
	C	RANGE DIRECTION, I.E., A POSITIVE CHANGE IN				
	C	THE FUZING DISTANCE, Q2, IS IN THE NEGATIVE				
	C	RANGE DIRECTION.				
520	C	22 IF (ITTG.EQ.1) FZTS = DMIN*TAN(FZAS)				
	C	Q2 = Q2 + Z2*FZTS + FZTM				
	C	RF = GR - Q2*CCG0				
	C	HF = GH + Q2*SIND				
	C	DF = GD				
	C	GO TO 85				
	C	BACKUP FUZING				
530	C	16 HF = 0.				
	C	IBKUP = 1				
	C	IF (CMEGA.EQ.0.) GO TO 5				
	C	IF (NVT.EQ.0) GO TO 17				
	C	B7 XK = RDM(1)				
	C	DO 65 K=1,NVT				
	C	KK = K				
	C	IF (XK.LE.P) T(K) GO TO 66				
	C	65 CONTINUE				
	C	66 HF = VHT(KK)				
	C	IF (HF.LE.HF) GO TO 24				
	C	HF = HF				
	C	17 RF = GR - (HF-GH)/TANO				
	C	DF = GD				
	C	GO TO 61				
	C	5 WRITE (6,*), "NC BACKUP FUZING FOR CMEGA = 0."				
	C	WRITE (6,*), "TRAJECTORY CLOSEST POINT OF APPROACH TO TARGET"				
	C	WRITE (6,*), "CENTER IS USED"				
	C	RF = 0.				
	C	DF = GD				
	C	HF = GH				
	C	GO TO 61				
	C	HEIGHT FUZING				
550	C					
	C	555 74 IF (SIND.EQ.0.) STOP 74				
	C	CALL BOXND (Z1,Z2)				
	C	HF = FZTM + Z1*FZTS				
	C	RF = RF + (GH-HF)*TANO				
	C	85 IF (OMEGA.EQ.0.) GO TO 24				
	C	IF (NVT.NE.0) GC TO 87				
	C	CHECK FGR FUZING POINT EELDOW GROUND				
565	C	565 24 IF (HF.GE.0.) GC TO 61				
	C	IF (CMEGA.EQ.0.) GO TO 61				
	C	RF = RF + HF/TANO				
	C	HF = 0.				
	C	PUT BURST POINT IN TARGET COORDINATE SYSTEM FOR				

PROGRAM ARP 73/74 QPT=1 FTN 4.8+508 03/13/81 08.29.30 PAGE 11

```

C      BLAST AND DIRECT HIT COMPUTATIONS.
C
 61 CALL ROTATE (RF,DF,DHAZ,-1.)
    BR = RF
    BD = DF
    BH = HF
    IF(NDBG,GE,1) WRITE (6,*)
    "BR,BD,BH AT STMT 61 * ",BR,BD,BH
    SET UP BLST VALUE FOR BLST VS. HGT
C
 575
    IF(NBLST,LE,C) GO TO 105
    DO 10 I=1,NBLST
    IF(HF,GT,HBLST(I)) GO TO 10
    BLST = BBLST(I)
    GO TO 105
 10 CONTINUE
    BLST = 0.
    WRITE (6,*)
    "HF EXCEEDS ALL HBLST, HF = ",HF
    GO TO 18
 105 IF(NDHT,EQ,0) GO TO 106
 590
    C DETERMINE DIRECT HIT FK
    C
    USE 2 POINTS TO DEFINE TRAJECTORY. BURST POINT
    (BR,BD,BH) AND POINT AT BR+10 (RBS,DBS,HBS).
    C IF AZIMUTH ATTACK ANGLE IS 90 DEGREES, SET
    RBS,DBS,HBS POINT AT ED+10.
    C (RPN,DPN,HPN) WILL BE BURST POINT, WITH OR
    C WITHOUT DIRECT HIT.
    C
    IPN IS PENETRATION INDEX (0 = NO PENETRATION,
    C N = BOX N PENETRATED)
    C
 595
    RPN = BR
    DPN = BD
    HPN = BH
    IF(ABS(DATA(16)),EQ,90) GO TO 95
    RBS = BR + 10.
    DBS = BD - 10.*TAN(DHAZ)
    HBS = BH - 10.*TANO/COS(DHAZ)
    GO TO 96
 600
    95 RBS = BR
    DBS = BD + 10.
    HBS = BH + 10.*TANO
    96 IPN = 0
    C
    C CHECK EACH ECX FOR PENETRATION
    C
 605
    IF(NDBG,EQ,1) WRITE (6,*)
    "OMEGA,RBS,DBS,HBS = ",OMEGA,RBS,DBS,HBS
    IF(NDBG,EQ,1) WRITE (6,*)
    "RF,DF,HF = ",RF,DF,HF
    IF(NDBG,EQ,1) WRITE (6,*)
    "CR,GD,GH = ",CR,GR,GO,GH
    DO 92 I=1,NDHT
    IF(BR,LT,EDH(I,1)) GO TO 92
    IF(DATA(16),NE,0.) GO TO 105
    IF(BD,LT,DDH(I,1),OR,BO,GT,DDH(I,2)) GO TO 92
    109 IF(GH,GT,HDH(I,2),AND,OMEGA,GE,0.) GO TO 92
 620
    92
    105
    109
 625
  
```

PROGRAM AFP 73/74 CPT=1 FTN 4.8+508 03/13/81 08.29.30 PAGE 12

```

IF(BH.LT.HDH(I,1).AND.OMEGA.EQ.0.) GO TO 92      006370
RDH1 = RDH(I,1)      006380
RDH2 = RDH(I,2)      006390
DDH1 = DDH(I,1)      005400
DDH2 = DDH(I,2)      006410
HDH1 = HDH(I,1)      006420
HDH2 = HDH(I,2)      006430
006440
630      C
IPEN = NUMBER OF STREES PENETRATED (MUST BE 0 OR 2)      006450
IF(AES(DATA(16)).EQ.90.) GC TO 102      006450
006460
635      C
IPEN = 0      006470
IF(AES(DATA(16)).EQ.90.) GC TO 102      006480
006490
C
CHECK RANGE SIDES      006500
C
DO 97 K=1,2      006510
RDHX = RDH1      006520
IF(K.EC.2) RDHX = RDH2      006530
CALL SEARCH (I,1,RDHX,DA,HA)      006540
IF(NCBG.EQ.2) WRITE (6,*)"IPEN,ROHX,DA,HA = ",IPEN,
1 RDHX,DA,HA      006550
006560
97 CONTINUE      006570
IF(IPEN.EQ.2) GC TO 92      006580
102 IF(DATA(16).EQ.6..OR.DATA(16).EQ.180.) GO TO 108      006590
006600
640      C
CHECK DEFLECTION SIDES      006610
C
DO 107 K=1,2      006620
DDHX = DDH1      006630
IF(K.EQ.2) DDXH = DDXH2      006640
CALL SEARCH (I,2,RA,DDHX,HA)      006650
IF(NCBG.EQ.2) WRITE (6,*)"IPEN,RA,DDHX,HA = ",IPEN,
1 RA,DDHX,HA      006660
IF(IPEN.EQ.2) GC TO 92      006670
107 CONTINUE      006680
108 IF(OMEGA.EQ.0.) GC TO 101      006690
006700
645      C
CHECK HEIGHT SIDES      006710
C
DO 117 K=1,2      006720
HDHX = HDH1      006730
IF(K.EQ.2) HDHX = HDH2      006740
CALL SEARCH (I,2,RA,DA,HDHX)      006750
IF(NCBG.EQ.2) WRITE (6,*)"IPEN,RA,DA,HDHX = ",IPEN,
1 RA,DA,HDHX      006760
IF(IPEN.EQ.2) GC TO 92      006770
117 CONTINUE      006780
101 IF(IPEN.EQ.1) STOP 117      006790
92 CONTINUE      006810
IF(IPEN.EQ.0) GC TO 106      006820
PKDH = PKCH + PKDHX      006830
006840
650      C
SET JP BURST COORDINATES (BR,BD,BH) FROM DIRECT HIT.      006850
006870
006880
655      C
BR = RPN      006890
BD = DPN      006910
BH = HPN      006920
006930

```

685 106 IF(BH.GE.C.) GO TO 37
IF(SMEGA.EQ.0.) STOP 106BR = BR + BH/TANO
BH = 0.C COMPUTE NEAR MISS BLAST KILL
C37 IF(NBLST.EQ.0) GO TO 90
IF(NDHT.EQ.0) GO TO 103

DO 104 I=1,NDHT

IBLST = 1
CALL BLAST (IBLST,BR,BLST,RDH,I)
CALL BLAST (IBLST,BD,BLST,CDH,I)
CALL BLAST (IBLST,BH,BLST,HDH,I)
IF(IBLST.EQ.1) GO TO 11

104 CONTINUE

GO TO 90
103 ISF = SQRT(BR*BR + BD*BD + (BH-TGTC)*(BH-TGTC))

IF(DIST.GT.BLAST) GO TO 90

11 PKBLST = PKBLST + PKBLX

C COMPUTE RADAR BLAST KILL
C90 IF(NCBG.EQ.2) WRITE (6,*),IPN,RPN,DPI,HPN,BR,BD,BH = ".
C IPN,RPI,DPI,HPN,BR,BD,BH
IF(NDR.EQ.0) GO TO 27

ERDR = BR-RDR(1)

DRDR = BD-EDF(2)

HRDR = BH-RDP(5)

RRDR = SQRT(BR*BR+BD*BD+CDRDR*CDRDR+HRDR*HRDR)

PKDR = 1.0
IF(XRDR.GT.RDR(4)) PKRDR = 1. - (QRDR-RDR(4))/(RDR(5)-RDR(4))

IF(RDR.GE.RDR(5)) PKRDR = 0.

007220

5004 FORMAT (1X,*BR,BD,BH = *,3(F6.1,*,*,1X))
27 IBX = 0
IROT = 0
IF(NCBG.GE.1) WRITE (6,5004) BR,BD,BH
IF(NH.EQ.0) GO TO 50

C COMPUTE PK DUE TO FRAGMENTATION (PKSAMIP)

715 C INTERPOLATE IN RANGE, DEFLECTION, HEIGHT & ANGLE TO
C GET FRAGMENTATION PK FROM PK GRIDS.
C
68 C
725 C
730 C
735 C
740 C007320
007330
007340
007350
007360
007370
007380
007390
007400
007410
007420
007430
007440
007450
007460
007470
007480
007490
007500ROTATE BLAST POINT FOR FRAGMENTATION PK
INTERPOLATION INTO ARP COORDINATE SYSTEM.
RECALL THAT PK GRIDS ARE IN PROJECTILE COORDINATE
SYSTEM.
CALL ROTATE (BR,3D,CHAZZ,1.)
IROT = 1
LOCATE HEIGHT BOUNDARIES

PROGRAM ARP 73/74 CPT=1 FTN 4.B+508 03/13/81 08.29.30 PAGE 14

```

C DO 20 I=1,NH
IH2 = 1
IF(BH.LE.HGT(I)) GO TO 25
20 CONTINUE
IH2 = 0
25 IH1 = IH2 - 1
IF((IH1.EQ.0)) IH1 = 1
IF((IH1.LT.0)) IH1 = NH
IF(NDBG.EQ.4) WRITE(6,*),IH1,IH2,NR,ND,RU,DU,BR,BD,BH =
C IH1,IH2,NR,ND,RU,DU,BR,BD,BH
31 CALL INTERP(BR,BD,BH,RGRD,DGRD,HGT,IH1,IH2,PKA,NR,ND,RU,DU,NH,007620
C NDBG)
PKSAMP = PKA
GO TO 4;
50 PKSAMP = 0.
C COMPUTE SPHERICAL COORDINATES TO BURST POINT (BR,BD,BH)
C FROM GROUND ZERO (0,0,0)
C SA1 = ANGLE OFF POSITIVE RANGE AXIS MEASURED
C CLOCKWISE
C SA2 = ANGLE OFF R-D PLANE MEASURED TOWARD POSITIVE
C SR = RANGE FROM BURST POINT TO (0,0,0)
C H-AXIS IN VERTICLE PLANE
41 IF(NDBG.EQ.4) WRITE(6,*),PK(FRAG) = "",PKSAMP
C GET BURST POINT BACK INTO TARGET COORDINATE
C SYSTEM IF IRGT = 1.
C IF(IRGT.EQ.1) CALL ROTATE(BR,BD,DHAZ,-1.)
BRR = BP.BR
BDD = BD*BD
BH = BH*BH
RR = BRR + BDD + BH
RR = SORT(RRR)
WRITE(6,*),BR,BD,BH,RR
ERBAR = BFEAR + CR
ERBAR2 = BRBAR2 + BRR
BDBAR = BDEAR + BD
BDBAR2 = BDBAR2 + BDD
ERBAR = ERBAR + BH
BHBAR2 = BHBAR2 + BH
RRBAR = RRBAR + RR
RRBAR2 = RRBAR2 + RRR
SA1 = PI/2.
SA2 = 0.
IF(BR.EQ.0.) GO TO 55
SA1 = ATAN2(BD,BR)
IF((SA1.LT.0.)) SA1 = 2.*PI + SA1
55 IF((BD.EQ.0. AND BR.EQ.0.)) GO TO 56
SA2 = ATAN(BH/SORT(BR*BR+BD*BD))
56 SA1 = SA1*360./(2.*PI)
SA2 = SA2*360./((2.*PI))
DO 57 I=1,12
ISA1 = I
IF((SA1.LT.ALPHA(I+1))) GO TO 58

```

PROGRAM ARP	73/74	CFT=1	FTN 4.0+508	03/13/81	08.29.30	PAGE	15
800							
	57	CONTINUE		008080	008090		
	58	D9 98 I=1,6		008100	008110		
		ISA2 = I		008120	008130		
		IF(ISA2.LT.BETA(I+1)) GO TO 99		008140	008150		
	98	CONTINUE		008160	008170		
	99	SR = SQRT(BR*BR + BD*BD + BH*BH)		008180	008190		
805		ISR = 0		008200	008210		
		DO 100 I=1,10		008220	008230		
	100	I1 = 1		008240	008250		
		IF(I.E..LT.10) I1 = 11		008260	008270		
		ISR = SR + 1		008280	008290		
810		IF(SR.E.T.RANGE(11)) GO TO 110		008300	008310		
	100	CONTINUE		008320	008330		
	110	IF(NDBG.EQ.6) WRITE (6,*,"ISA1,ISA2,ISR = ",ISA1,ISA2,ISR)		008340	008350		
		IF(NDBG.EQ.6) WRITE (6,*,"SA1,SA2,SR = ",SA1,SA2,SR)		008360	008370		
815	C	STORE PK'S ACCORDING TO SPHERICAL COORDINATES		008380	008390		
	C	IKS(ISA1,ISA2,ISR) = IKS(ISA1,ISA2,ISR) + 1		008400	008410		
	C	SUM PK'S OVER ALL SAMPLES		008420	008430		
820	C	IF(NDBG.GT.0) WRITE (6,*,"PKR,PKR,PKR,PKB = ",PKSAM,PKRDR,PKDH)		008440	008450		
	C	C,PKBLST		008460	008470		
		PKSAM = PKBASE + PKSAM		008480	008490		
		PKRDR = PKRADR + PKRDH		008500	008510		
		PKDHIT = PKDHIT + PKDH		008520	008530		
825		PKBLST = PKBLST + PKBLST		008540	008550		
		PKSAM = 1. - (1.-PKSAM)*(1.-PKRDR)*(1.-PKDH)*(1.-PKBLST)		008560	008570		
		PKS(ISA1,ISA2,ISR) = PKS(ISA1,ISA2,ISR) + PKSAM		008580	008590		
		PKTOT = PKTOT + PKSAM		008600	008610		
		PKTOT2 = PKTOT2 + PKSAM		008620	008630		
		IF(NDBG.GE.1) WRITE (6,3003) PKSAM		008640	008650		
830		3003 FORMAT (5X,*SAMPLE PK = *,F6.4)		008660	008670		
		IF(NERT.EQ.1) GO TO 1		008680	008690		
		IF(MOD(ISIM,10).NE.0) GO TO 1		008700	008710		
835		PKPRNT = ISIM		008720	008730		
		PKPRNT = PKTOT.PKPRNT		008740	008750		
		WRITE (6,*,"NO. SIMULATIONS, PK = ",ISIM,PKPRNT)		008760	008770		
		GO TO 1		008780	008790		
	18	NCT = NCT + 1		008800	008810		
B40	C	1 CONTINUE		008820	008830		
	C	DISPLAY FINAL RESULTS		008840	008850		
	C	I(NPRT.GT.0) GC TG 79		008860	008870		
		WRITE (6,2002)		008880	008890		
		WRITE (6,*,"FINAL RESULTS")		008895	008900		
845		3000 FORMAT (/,1X,*PK = *,F6.4,2X,*PKSD = *,F6.4,2X,*NSAMP = *,16,/)		008910	008920		
	79	XSAM = NSAMP		008930	008940		
		PKBAR = PKTOT/XSAM		008950	008960		
		PKBASE = PKBASE/XSAM		008970	008980		
		PKRADR = PKRADR/XSAM		008990	009000		
		PKDHIT = PKDHIT/XSAM		009010	009020		
		PKBLST = PKBLST/XSAM		009030	009040		
		PK(ILUP) = PKBASE		009050	009060		
855		PKR(ILUP) = PKRADR		009070	009080		

```

PKD(ILUP) = PKCHIT
PKG(ILUP) = PKBAR
PKBL(ILUP) = PKBLT
XSAMP = NSMP - NCT
IF(NSMP.EQ.NCT) XSAMP = 1.
XSMP = XSAMP-1.
IF(XSAMP.EQ.0.) XSAMP = 1.
BRG(ILUP) = BRBAR/XSAMP
BDG(ILUP) = BDEAR/XSAMP
BHG(ILUP) = BHBAR/XSAMP
RRG(ILUP) = RRBAR/XSAMP
BxSG(ILUP) = SQR((BRBAR2 - XSAMP*BHG(ILUP))*BRG(ILUP))/XSMP
BDSG(ILUP) = SQR((BDEAR2 - XSAMP*BDG(ILUP))*BDG(ILUP))/XSMP
BHSG(ILUP) = SQR((BHBAR2 - XSAMP*BHG(ILUP))*BHG(ILUP))/XSMP
RRSG(ILUP) = SQR((RRBAR2 - XSAMP*RRG(ILUP))*RRG(ILUP))/XSMP
IF(NCT.NE.0) WRITE(6,2004) NCT,NSMP,CEP(ILUP)
C * SIMULATIONS, * ,/2X,* GUIDANCE CEP = *,F6.2)
2004 FORMAT(2X,*PROJECTILE OR FUZING DUDS = *,14,* OUT OF *,14,
C * SIMULATIONS, * ,/2X,* GUIDANCE CEP = *,F6.2)
IF(IPT.GT.0) GO TO 69
IF(PKSD.LT.0.) PKSD = 0.
IF(PKSD.LT.0.) PKSD = 0.
PKSD = SQR(PKSD)
WRITE(6,3002) PKBAR,PKSD,NSMP
WRITE(6,2002)
WRITE(6,*) "DO YOU WANT PK VS R, ALPHA, BETA? "
READ(5,1001) ANS
IF(ANS.NE.YES) GO TO 44
C
C          PK VS R, ALPHA, BETA, WHERE ALPHA IS AZIMUTH ANGLE
C          MEASURED FROM POSITIVE RANGE AXIS TOWARD POSITIVE
C          DEFLECTION AXIS (0 TO 360). BETA IS ELEVATION ANGLE
C          MEASURED FROM NEGATIVE HEIGHT AXIS TO POSITIVE
C          HEIGHT AXIS (0 TO 90).
C
C          WRITE(6,2001)
C          WRITE(6,*) " PK           R           ALPHA           BETA"
C          WRITE(6,*) "----- ----- ----- ----- ----- ----- -----"
C          DO 49 I=1,10
C          DO 49 J=1,12
C          DO 49 K=.6
C          IF(IKS(J,K,I).EQ.0) GO TO 49
C          XIKS = IKs(J,K,I)
C          PKS(J,K,I) = PKS(J,K,I)/XIKS
C          CONTINUE
C          DO 45 I=1,10
C          XI = 0.
C          RSUM(I) = 0.
C          RANG = RANGE(I)
C          DO 47 J=1,12
C          DO 47 K=1,6
C          RPK = PMS(J,K,I)
C          XIKS = IKs(J,K,I)
C          XI = XI + XIKS
C          RSUM(I) = RSUM(I) + XIKS*RPK
C          IF(RPK.GT.0.) WRITE(6,3004) RPK,RANG,ALPHA(J),ALPHA(J+1),BETA(K),
C          CBETA(K+1)
C          3004 FORMAT(1X,F6.4,2X,F5.1,2(2X,F6.1,* - *,F6.1))

```

PROGRAM ARP	73/74	OPT=1	FTN 4.8+508	03/13/81	08.29.30	PAGE 17
47	CONTINUE					
915	IF(XI.EQ.0.) GO TO 45					
	RSUM(I) = RSUM(I)/XI					
45	CONTINUE					
	WRITE (6,2002)					
	WRITE (6,*), "AVG PK VS. R"					
	WRITE (6,*), _____					
920	DO 43 I=1,10					
	R = RANGE(I)					
	I=(RSUM(I).EQ.0.) GO TO 43					
	WRITE (6,3001) RSUM(I),R					
43	CONTINUE					
925	3001 FORMAT (1X,F6.4,4X,F5.1)					
	WRITE (6,2002)					
	C CHECK FOR ANOTHER CASE					
	C 44 WRITE (6,2001)					
930	69 CONTINUE					
	C DISPLAY RESULTS FOR EACH GUIDANCE ERROR					
	C IF(NPRI.GT.0) WRITE (6,2002)					
935	FZTM = DATA(2)					
	OMEGD = DATA(7)					
	OMGSD = DATA(19)					
	FZAD = DATA(1)					
	FZASD = DATA(5)					
940	WRITE (6,2006) GMEGD,OMGSD,FZAMD,FZASD,FZTM,FITS,DHAZ,NSMP					
	2006 FORMAT (/,*5X,*RECORD FOR FOLLOWING CONDITIONS - *,//,					
	C12X,*ITEM*,13X,*MEAN*,4X,*STD DEV*,//,					
	C10X,*ELEVATION*,4X,2F10.4,/,*10X,*FUZE ANGLE*,3X,2F10.4,/,					
	C10X,*LINEAR FUZE*,2X,2F10.4,/,*10X,*AZIMUTH*,*F10.4,/.					
	C10X,*SAMPLE SIZE - *,15,/*					
	WRITE (6,2003) GMER,GMD,GMH					
	WRITE (6,2012)					
	2012 FORMAT (/,*5X,*ERROR DATA*,17X,*PK*,3X,					
	C*PKFRAG PKRADR PKDHIT PKBLST*)					
950	2005 FORMAT (5X,*HOMING POINT COORDINATES (R,D,H) = *,					
	C 2(F6.1,*,*), F6.1)					
	DO 72 I=1,NLOOP					
	IF(NCEP.EQ.0) WRITE (6,2007) SDD(I),SDH(I),PKG(I)					
	C,PK(I),PKR(I),PKD(I),PKBL(I)					
	IF(NCEP.EQ.1) WRITE (6,2008) CEP(I),PKG(I)					
	C,PK(I),PKR(I),PKD(I),PKBL(I)					
955	72 CONTINUE					
	WRITE (6,2002)					
	DO 26 I=1,NLOOP					
960	26 WRITE (6,1004) CEP(I),RRG(I),BRG(I),RRSG(I),BDSG(I)					
	C,BHG(I),BHSG(I)					
	1003 FORMAT (/,*5X,*BURST STATISTICS (MEAN, STD DEVIATION*,//,*CEP*,					
	C,4X,*BURST RANGE*,7X,*RANGE*,8X,*DEFLECTION*,7X,*HEIGHT*)					
	1004 FORMAT (1X,F4.1,4,(2X,F6.2,1X,F6.2))					
	2007 FORMAT (5X,*SD (D,H) -,					
	C2(F4.1,*,*),1X,5F7.4)					
	2008 FORMAT (5X,*CEP,-*,F4.1,					
	C14X,5F7.4)					

PROGRAM ARP

73/74 OPT=1

FTN 4.8+503

03/13/81

08.29.30

PAGE 18

```

970      WRITE (6,*), "DO YOU WISH TO RUN ANOTHER CASE? "
         READ (5,1001) ANS
         IF (ANS.EQ.YES) GO TO 15
         STOP
         END

```

SYMBOLIC REFERENCE MAP (R=2)

ENTRY POINTS 3506 ARP	DEF LINE 1	REFERENCES	RELOCATION	REFS	199	DEFINED 166	199	DEFINED 173	197	221	226
VARIABLES	SN	TYPE				DEFINED	164	217	218		
10232 A		REAL		REFS	168	217	168	217	218		
10227 AAAA		REAL		REFS	486	487	486	487	488		
10373 AB		REAL		REFS	480	486	480	486	487		
10371 AB2		REAL		REFS	7	7	7	7	7		
31267 ALPHA		REAL	ARRAY	REFS	3	168	168	175	175		
10455 ANAM		REAL	ARRAY	DEFINED	21						
10376 ANG		REAL		REFS	496	497	496	497	502		
10217 ANS		REAL		DEFINED	494	495	494	495	506		
10233 B		REAL		REFS	55	151	55	151	506		
132 BBLST		REAL		DEFINED	54	149	54	149	169		
10372 BB2		REAL		REFS	199	199	199	199	197		
6 BD		REAL		REFS	15	584	15	584	584		
				DEFINED	486	486	486	486	486		
				REFS	13	577	13	577	606		
				DEFINED	2*702	708	2*702	708	708		
				REFS	2*774	776	2*774	776	781		
				DEFINED	575	683	575	683	683		
				REFS	781	864	781	864	864		
				REFS	782	868	782	868	868		
				REFS	776	868	776	868	868		
				REFS	11	2*868	11	2*868	961		
				RCFS	11	961	11	961	961		
				FEFS	7	802	7	802	802		
				FEFS	487	502	487	502	502		
				FEFS	13	577	13	577	607		
				REFS	685	687	685	687	696		
				SRCH	745	751	745	751	753		
				DEFINED	576	684	576	684	684		
				REFS	783	865	783	865	865		
				REFS	784	869	784	869	869		
				REFS	12	2*869	12	2*869	961		
				REFS	776	784	776	784	784		
				REFS	12	961	12	961	961		
				REFS	271	272	271	272	275		
				DEFINED	270	274	270	274	274		
				REFS	13	577	13	577	605		
				SRCH	696	2*702	696	2*702	708		
				DEFINED	772	2*773	772	2*773	778		
				REFS	2*804	DEFINED	2*804	DEFINED	682		

009790
009800
009810
009820
009830

009720
009710
009700
009690
009680

02*804
02*803
02*802
02*801

02*800
02*799
02*798
02*797

02*796
02*795
02*794
02*793

02*792
02*791
02*790
02*789

PROGRAM	ARP	VARIABLES	SN	TYPE	RELOCATION
73/74	OPT=1	10315	BRRBAR	REAL	
		10316	BRRBAR2	REAL	
		10413	BRDR	REAL	ARRAY
		31516	BRG	REAL	ARRAY
		10425	BRR	REAL	ARRAY
		31530	BRRSG	REAL	ARRAY
		10365	CB	REAL	ARRAY
		31325	CEP	REAL	ARRAY
		10334	COSD	REAL	
		10340	D	REAL	
		10404	DA	REAL	
		10537	DATA	REAL	ARRAY
		3	DBS	REAL	
		31446	DDH	REAL	ARRAY
		10406	DDHX	REAL	
		16	DDH1	REAL	
		17	DDH2	REAL	
		10347	DF	REAL	
		10357	DGLT	REAL	
		30735	DGRD	REAL	ARRAY
		10252	DHAZ	REAL	
		10412	DIST	REAL	
		10275	DL	REAL	
		10342	DMIN	REAL	
		12	DPN	REAL	
		10414	DRDR	REAL	
		10215	DTE	REAL	
		145	DU	REAL	
		10265	DUDR	REAL	
		6522	END	REAL	
		10234	FZAM	REAL	
		10453	FZAMD	REAL	
		10240	FZAS	REAL	
		10454	FZASD	REAL	
		10375	FZASX	REAL	
		10235	FZTF	REAL	
		10242	FZTS	REAL	
		10377	GAMMA	REAL	
		10344	GD	REAL	
		10367	GDL	REAL	
		10345	GH	REAL	
		10370	GHL	REAL	ARRAY
		21	GLTR	REAL	
		65	GMD	REAL	
		RDWRT			

	FTN 4.8+503	03/13/81	08.29.30	PAGE
REFS	779	863	DEFINED	345
REFS	780	867	DEFINED	346
REFS	2*714	DEFINED	711	779
REFS	11	2*867	961	780
REFS	776	780	DEFINED	773
REFS	11	961	DEFINED	863
REFS	469	3*486	487	B67
REFS	646	8	DEFINED	468
REFS	2*375	395	524	955
REFS	391	392	DEFINED	961
REFS	645	647	670	955
REFS	936	3	175	961
REFS	239	240	181	961
REFS	250	252	182	961
REFS	261	265	185	961
REFS	282	283	186	961
REFS	937	938	187	961
REFS	10	113	188	961
REFS	2*626	631	189	961
REFS	658	659	190	961
REFS	13	656	191	961
REFS	550	573	192	961
REFS	440	444	193	961
REFS	429	438	194	961
REFS	386	5	195	961
REFS	772	940	196	961
REFS	703	DEFINED	197	961
REFS	302	DEFINED	198	961
REFS	522	DEFINED	199	961
REFS	113	683	200	961
REFS	2*714	DEFINED	201	961
REFS	46	48	202	961
REFS	166	366	203	961
REFS	15	751	204	961
REFS	940	3*495	205	961
REFS	245	DEFINED	206	961
REFS	940	246	207	961
REFS	494	DEFINED	208	961
REFS	494	495	209	961
REFS	523	558	210	961
REFS	523	558	211	961
REFS	506	507	212	961
REFS	402	419	213	961
REFS	622	DEFINED	214	961
REFS	479	487	215	961
REFS	403	419	216	961
REFS	551	622	217	961
REFS	479	487	218	961
REFS	6	15	219	961
REFS	439	DEFINED	220	961
REFS	15	385	221	961

PROGRAM ARP	73/74	OPT=1	RELOCATION		FTN: 4.6+50B	C3/13/81	08.29.30	PAGE	20
VARIABLES	SN	TYPE							
10337	GMDR	REAL			REFS	386	394	DEFINED	385
66	GMH	REAL			REFS	15	395	946	
64	GMR	REAL			REFS	15	384	946	
10336	GRMR	REAL			REFS	386	393	DEFINED	384
10343	GR	REAL			REFS	401	419	474	478
10366	GRL	REAL			REFS	479	487	DEFINED	474
10362	GRMAX	REAL			REFS	510	DEFIN	434	511
10341	H	REAL			REFS	391	392	395	
10405	HA	REAL			REFS	646	647	656	659
137	HBLST	REAL	ARRAY	RDWRT	REFS	15	583	DEFINED	273
4	HBS	REAL	ARRAY	SRCH	REFS	13	626	DEFINED	611
31460	HDH	REAL	ARRAY		REFS	10	175	185	292
10410	HDHX	REAL			REFS	297	627	628	633
20	HDH1	REAL			REFS	670	671	DEFINED	668
21	HDH2	REAL			REFS	13	668	DEFINED	533
10350	HF	REAL			REFS	13	669	DEFINED	634
10462	HFX	REAL			REFS	541	543	559	565
10360	HGLT	REAL	ARRAY	RDWRT	REFS	588	621	DEFINED	403
0	HGT	REAL		SRCH	REFS	558	568		
13	HPN	REAL		SRCH	REFS	541	542	DEFINED	446
10415	HRDR	REAL	INTEGER		REFS	446	478	475	439
10326	1	REAL			REFS	6	15	731	
					REFS	13	684	708	DEFINED
					REFS	2*71:4	DEFIRED	71:3	607
					REFS	158	121	2*182	
					REFS	291	292	295	2*289
					REFS	2*312	326	329	309
					REFS	2*626	627	626	
					REFS	634	646	658	624
					REFS	745	797	798	624
					REFS	897	2*598	902	632
					REFS	921	922	923	635
					REFS	200	157	180	744
					REFS	300	305	308	744
					REFS	694	743	795	808
					REFS	952	965	976	896
					REFS	467	532	595	2*915
					REFS	696	697	698	2*909
					REFS	719	DEFINED	25	
					REFS	435	436	509	DEFINED
					REFS	251	419	439	439
					REFS	437	438	428	428
					REFS	512	427	429	427
					REFS	419	420	DEFINED	418
					REFS	749	750	751	748
					REFS	748	751	752	749
					REFS	810	DEFINED	730	747
					REFS	5	DEFINED	807	747
					REFS	5	817	896	808
					REFS	333	617	837	807
					REFS	351	352	854	858
					REFS				

PROGRAM	ARP	73/74	CPT=1	RELOCATION	REFS	FTN 4.8+508	03/13/81	08.29.30	PAGE	22
VARIABLES	SN	TYPE			DEFINED	508	510	523	524	525
16352	02	REAL			REFS	413	507	523	524	525
10225	P1	REAL			REFS	358	497	502	506	507
31364	PK	REAL	ARRAY		REFS	795	156	955	955	954
10424	PKA	REAL			REFS	753	953	955	955	954
10442	PKBAR	REAL			REFS	857	2*875	878	878	849
10305	PKBASE	REAL			REFS	823	850	854	854	823
31422	PKBL	REAL	ARRAY		REFS	9	953	955	955	850
10327	PKBLST	REAL			REFS	704	621	826	827	704
10306	PKBLT	REAL			REFS	826	653	858	858	853
10237	PKBLX	REAL			REFS	279	704	DEFINED	338	826
31410	PKD	REAL	ARRAY		REFS	9	953	955	955	279
10326	PKDH	REAL			REFS	678	821	825	827	825
10304	PKDHIT	REAL			REFS	825	852	856	856	878
10236	PKDHX	REAL			REFS	278	678	DEFINED	336	852
31313	PKG	REAL	ARRAY		REFS	8	953	955	955	857
26541	PKM	REAL	ARRAY		REFS	4	328	406	406	259
10256	PKPF	REAL			REFS	349	350	637	637	349
10440	PKPRNT	REAL			REFS	836	953	955	955	350
31376	PKR	REAL	ARRAY		REFS	9	824	851	855	851
10303	PKRADR	REAL			REFS	824	824	827	827	829
10330	PKRDR	REAL			REFS	821	824	827	827	716
25221	PKS	REAL	ARRAY		REFS	717	4	828	898	906
10325	PKSAM	REAL			REFS	898	767	821	823	828
10444	PKSD	REAL			REFS	831	359	755	757	829
10307	PKTOT	REAL			REFS	876	877	878	878	877
10310	PKTC ₁ ₂	REAL			REFS	829	836	849	849	829
10621	PK1	REAL			REFS	830	875	DEFINED	340	830
10621	PK2	REAL			REFS	3	319	753	753	
57	PVT	REAL			REFS	7	15	2*306	538	DEFINED
10450	R	REAL			REFS	923	DEFINED	921	921	306
10407	RA	REAL			REFS	658	659	670	671	
10446	RANG	REAL			REFS	910	DEFINED	903	903	
31337	RANGE	REAL			REFS	8	810	903	921	
2	RBS	REAL			REFS	13	620	DEFINED	609	
31434	RDH	REAL	ARRAY		REFS	10	175	185	202	
10403	RDHX	REAL			REFS	624	629	630	696	
14	RDH1	REAL			REFS	646	647	647	647	
15	RDH2	REAL			REFS	13	644	644	644	
125	RDR	REAL			REFS	13	645	645	645	
10346	RF	REAL			REFS	15	711	712	713	
10356	RGLT	REAL			REFS	559	567	573	574	
30225	RGRD	REAL	ARRAY		REFS	401	524	543	543	
10447	RPK	REAL			REFS	440	444	446	446	
11	RPN	REAL			REFS	428	437	478	478	
10431	RR	REAL			REFS	5	519	753	753	
10311	RRBAR	REAL			REFS	909	2*910	DEFINED	906	
10312	RRBAR2	REAL			REFS	13	682	708	708	
10416	RRDR	REAL			REFS	778	785	DEFINED	777	
31472	RRG	REAL	ARRAY		REFS	785	866	866	866	
10274	RRNG	REAL			REFS	787	870	870	870	

PROGRAM	ARP	73/74	CPT=1	RELOCATION		FTN 4.8+508	03/13/81	08.29.30	PAGE	23
VARIABLES	SN	TYPE								
10430 RRR		REAL		ARRAY	REFS	777	786	DEFINED	776	
31504 RRSQ		REAL		ARRAY	REFS	11	961	DEFINED	870	
31352 RSUM		REAL		ARRAY	REFS	8	909	DEFINED	922	923
144 RU	REAL	REAL		RDWRT	REFS	902	909	915		
10432 SA1	REAL	REAL		RDWRT	REFS	15	751	753		
10433 SA2	REAL	REAL		RDWRT	REFS	2*791	794	798	813	DEFINED
67 SDD	REAL	REAL		RDWRT	REFS	791	794	813	DEFINED	787
101 SDH	REAL	REAL		RDWRT	REFS	15	351	953	802	790
10321 SIGD	REAL	REAL		RDWRT	REFS	392	394	351	813	DEFINED
10322 SIGH	REAL	REAL		RDWRT	REFS	392	393	395	802	795
10333 SINO	REAL	REAL		RDWRT	REFS	375	393	467	813	DEFINED
10436 SR	REAL	REAL		RDWRT	REFS	372	813	804	813	DEFINED
10262 SPNG	REAL	REAL		RDWRT	REFS	810	299	265	298	
10335 TANO	REAL	REAL		RDWRT	REFS	298	543	559	611	
10364 TANOX	REAL	REAL		RDWRT	REFS	466	543	559	611	
10254 TGTC	REAL	REAL		RDWRT	REFS	374	375	915	804	DEFINED
10216 THE	REAL	REAL		RDWRT	REFS	474	466	467	298	
10230 VALUE	REAL	REAL		RDWRT	REFS	2*702	47	48	265	
1014 VTHT	REAL	REAL		RDWRT	REFS	170	222	222	298	
10354 XGLT	REAL	REAL		RDWRT	REFS	6	15	15	312	
10276 XI	REAL	REAL		RDWRT	REFS	426	425	425	301	
78 10445 XIKS	REAL	REAL		RDWRT	REFS	302	908	914	915	DEFINED
10400 XX	REAL	REAL		RDWRT	REFS	908	908	909	897	907
11 XOMG	REAL	REAL		RDWRT	REFS	538	535	535	897	
10273 XRNG	REAL	REAL		RDWRT	REFS	6	15	312	897	
10441 XSAMP	REAL	REAL		RDWRT	REFS	297	299	302	293	295
10443 XSMP	REAL	REAL		RDWRT	REFS	898	850	851	853	863
6521 YES	REAL	REAL		RDWRT	REFS	664	866	867	868	870
10331 Z1	REAL	REAL		RDWRT	REFS	848	859	860	869	875
10332 Z2	REAL	REAL		RDWRT	REFS	862	867	868	870	875
FILE NAMES	MODE									
0 INPUT										
410 OUTPUT										
1020 TAPE1	FMT									
1430 TAPE2					WRITES	182	184	READS	197	MOTION
2040 TAPE3					NOTION	186	203		152	
2450 TAPE4					NOTION	153			162	
0 TAPE5	FMT				NOTION	154			179	
410 TAPE6	MIXED				READS	155				
					READS	54				
					WRITES	149				
					WRITES	30				
					WRITES	32				
					WRITES	45				
					WRITES	61				
					WRITES	69				
					WRITES	78				
					WRITES	87				
					WRITES	96				

PROGRAM	ARP	73/74	OPT=1			FTN 4.8+508	03/13/81	08.29.30	PAGE	24
FILE NAMES										
ACOS	REAL	105	107	108	109	110	111	112	113	114
ACOS	REAL	115	116	117	118	119	120	121	122	123
ALOG	REAL	124	125	126	127	128	129	138	139	140
ATAN	REAL	141	147	148	163	173	188	191	192	199
ATAN2	REAL	201	206	216	226	281	419	440	445	446
BLAST		487	508	546	547	548	577	588	620	621
BOXING		622	647	659	671	708	721	751	767	812
CONNEX		813	821	831	837	845	846	871	878	879
COS	REAL	680	690	891	892	910	917	918	919	923
DATE	REAL	926	929	934	940	946	947	953	955	956
EXP	REAL	956	961	970	MOTION	778	29			
GRIDS										
INTERP										
RDM	REAL	1	1	1	1	1	1	1	1	1
RDOMIN		1								
RDOMOUT		1								
READ		8								
ROTATE		8								
SEARCH		16								
SIN										
SQRT	REAL	1	1	1	1	1	1	1	1	1
TAN	REAL	1	1	1	1	1	1	1	1	1
TIME		6								
WRITE										
3080 TAPES FREE VARIABLES USED AS FILE NAMES, SEE ABOVE										
EXTERNALS	TYPE	ARGS	REFERENCES							
ACOS	REAL	1	LIBRARY 486							
ALOG	REAL	1	LIBRARY 299							
ATAN	REAL	1	LIBRARY 793							
ATAN2	REAL	2	LIBRARY 156							
BLAST		5	696	697	698	557				
BOXING		2	370	391	28					
CONNEX		1	LIBRARY 373	611						
COS	REAL	1	LIBRARY 46							
DATE	REAL	1	LIBRARY 302							
EXP	REAL	8	319							
GRIDS		16	753	366	408	426	495	535		
INTERP		1	134							
RDM	REAL	1	133							
RDOMIN		1								
RDOMOUT		1								
READ		8	175	229						
ROTATE		4	386	444	573	738	772			
SEARCH		5	646	658	670					
SIN	REAL	1	LIBRARY 372	2*507						
SQRT	REAL	1	LIBRARY 293	392	480	702	714	777	793	804
TAN	REAL	1	LIBRARY 868	869	870	877				B67
TIME		1	522	610						
WRITE		6	185	202						
INLINE FUNCTIONS										
ABS	REAL	1	INTRIN 240	DEF LINE REFERENCES		275	608	639		
AMIN1	REAL	0	INTRIN 251		246					
IABS	INTEGER	1	INTRIN 264		292	297				
ISIGN	INTEGER	2	INTRIN 263							
MOD	INTEGER	2	INTRIN 834							
SIGN	REAL	2	INTRIN 2*288		2*289					
STATEMENT	LABELS	DEF LINE	REFERENCES							
5725	1	840	837	833	834	838				
4226	2	228	215	224						
4231	3	229	218							
4222	4	225	220							
5103	5	546	533							
4156	6	201	198							
4057	7	163	171							
0	8	199	196							

STATEMENT	LABELS	DEF LINE	REFERENCES
4105	9	179	230
5157	10	586	582
5423	11	704	699
4141	12	192	187
0	13	214	213
4103	14	175	166
4005	15	134	972
5053	16	531	408
5075	17	543	534
5723	18	839	366
0	20	746	496
4704	21	440	589
5034	22	522	420
4162	23	203	190
5131	24	565	541
5500	25	748	745
0	26	961	960
5454	27	719	710
0	28	312	311
0	31	753	
5375	37	692	685
5517	41	924	920
6157	43	929	882
6163	44	916	900
6137	45	913	904
0	47	313	905
4466	48	893	
6065	49	757	922
5516	50	158	722
0	51	334	731
0	52	330	
4076	53	172	331
4001	54	130	332
5565	55	55	168
5575	56	792	
0	57	794	895
5610	58	799	894
4456	59	800	895
4001	60	798	
5136	61	130	896
0	65	307	
5071	66	306	674
0	72	308	
4447	67	309.	
6165	69	573	
0	70	539	
5071	66	540	
0	72	957	
5115	74	556	
4645	75	424	
4671	76	433	
4674	77	435	
4471	78	323	
5735	79	318	
4035	80	848	
4117	81	152	
4234	82	183	
0	83	231	
		232	

STATEMENT	LABELS	PROGRAM ACP	7S/74	GPT=1	FTN 4.6+5CB	03/13/61	08.29.30	PAGE	26
5531	84		513	435	51C				
5127	85		521	242C	527				
4562	86		424	514					
5257	87		535	561					
4226	88		147	136					
4175	89		209	235					
5425	90		708	692	701	703			
5354	92		676	623	624	626			
4324	94		275	271					
5207	95		613	603					
5215	96		616	612					
0	97		649	643					
0	98		600	600					
5617	99		804	602					
0	100		811	802					
5351	101		675	653					
5304	102		651	639					
5414	103		702	693					
0	104		700	694					
5165	105		599	581					
5366	106		685	590					
0	107		652	650					
5530	108		653	651					
5242	109		627	625					
5541	110		812	310					
0	111		302	300					
4415	115		298	285					
4407	116		296	285					
0	117		667	657					
4405	118		295	288					
0	119		292	290					
7214	1000	FMT	165	154	182	184	197	217	
7320	1001	FMT	219	54	149	189	207	681	971
7267	1002	FAT	260	193	261				
10125	1003	FAT	353	355					
10142	1004	FAT	965	967					
7326	2000	FMT	227	173	226				
6577	2001	FMT	51	890	929				
6574	2002	FMT	5	30	34	45	52	56	879
10050	2003	FMT	934	934					
7650	2004	FMT	950	946					
6567	2005	FMT	372	871					
10000	2006	FMT	49	48					
10145	2007	FAT	941	940					
10153	2008	FMT	966	953					
10n40	2012	FMT	958	955					
7333	2100	FMT	448	947					
7751	3001	FMT	847	878					
7614	3003	FMT	925	923					
7726	3004	FMT	832	831					
7370	5003	FMT	912	910					
7524	5004	FMT	448	419					
81			713	721					
LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES				
4052	51	1	157	158	INSTACK				
4370	53	2	167	172	GPT	EXITS			

LOGPS	LABEL	INDEX	FROM-TO	OPT=1	LENGTH	PROPERTIES	EXT REFS	EXITS
4110	B1	I	180 163		126	INSTACK	EXT REFS	
4145	B	I	196 199		118	INSTACK	EXT REFS	
4201	1,3	I	213 214		28	OPT	EXT REFS	NOT INNER
4205	2	I	215 228		248	INSTACK	EXT REFS	
4214	4	J	220 225		118	INSTACK	EXT REFS	
4237	83	I	231 232		28	OPT	EXT REFS	
4354	116	I	286 298		368	INSTACK	EXT REFS	NOT INNER
4372	119	J	290 292		68	INSTACK	EXT REFS	
4525	111	I	300 302		78	INSTACK	EXT REFS	
4443	68	I	305 306		38	INSTACK	EXT REFS	
4453	60	I	308 309		28	INSTACK	EXT REFS	
4462	28	I	311 312		38	INSTACK	EXT REFS	
4472	69	I,UF	323 930	1476B	38	INSTACK	EXT REFS	NOT INNER
4475	70	I	327 329		20B	INSTACK	EXT REFS	NOT INNER
4502	52	I	330 334		158	INSTACK	EXT REFS	NOT INNER
4503	52	J	332 334		38	INSTACK	EXT REFS	NOT INNER
4511	52	K	357 840	1165B	68	INSTACK	EXT REFS	NOT INNER
4543	1	ISIM	435 513		1378	INSTACK	EXT REFS	NOT INNER
4675	84	IGL	536 539		68	INSTACK	EXT REFS	NOT INNER
5063	65	K	582 586		108	OPT	EXT REFS	NOT INNER
5152	10	I	623 676		1248	OPT	EXT REFS	NOT INNER
5233	92	I	643 649		15B	OPT	EXT REFS	NOT INNER
5265	97	K	655 662		178	OPT	EXT REFS	NOT INNER
5311	107	K	667 674		178	OPT	EXT REFS	NOT INNER
5332	117	K	694 700		146	OPT	EXT REFS	NOT INNER
5400	104	I	743 746		68	INSTACK	EXT REFS	NOT INNER
5471	20	I	796 799		68	INSTACK	EXT REFS	NOT INNER
56C2	57	I	800 803		68	INSTACK	EXT REFS	NOT INNER
5611	98	I	806 811		138	OPT	EXT REFS	NOT INNER
5626	100	I	893 899		238	OPT	EXT REFS	NOT INNER
6052	49	J	894 899		17B	INSTACK	EXT REFS	NOT INNER
6053	49	K	895 899		58	INSTACK	EXT REFS	NOT INNER
6062	49	K	900 916		448	INSTACK	EXT REFS	NOT INNER
6076	45	I	904 913		21B	INSTACK	EXT REFS	NOT INNER
6103	47	J	905 913		268	INSTACK	EXT REFS	
6104	47	K	920 924		128	INSTACK	EXT REFS	
6150	43	I	952 957		40B	INSTACK	EXT REFS	
6211	72	I	960 961		24B	INSTACK	EXT REFS	
6255	26	I						
COMMON	BLOCKS	LENGTH						
	SRCH	18						
	RDWT	102						

STATISTICS
 PROGRAM LENGTH 30053E 12331
 BUFFER LENGTH 27078 1479
 CM LABELED COMMON LENGTH 1705 120
 52000B CM USED

SUBROUTINE ROTATE 73/74 CPT=1
 1 C SUBROUTINE ROTATE (R,D,PHI,SIGNX)
 C ROTATES COORDINATE SYSTEM FROM TARGET SYSTEM
 C TO PROJECTILE SYSTEM OR VICE VERSA, DEPENDING
 C ON THE VALUE OF SIGNX (+1 = TG PROJECTILE SYSTEM,
 C AND -1 = TG TARGET SYSTEM).
 5 C
 C RT = R
 C R = R*COS(PHI) - SIGNX*D*SIN(PHI)
 C D = D*COS(PHI) + SIGNX*R*T*SIN(PHI)
 C END
 10 C
 C 009840
 C 009850
 C 009860
 C 009870
 C 009880
 C 009890
 C 009900
 C 009910
 C 009920
 C 009930
 C 009940

SYMBOLIC REFERENCE MAP (R=2)
 ENTRY POINTS DEF LINE REFERENCES
 2 ROTATE 1 1:
 VARIABLES SN TYPE RELOCATION
 0 D REAL F.P.
 0 PHI REAL F.P.
 0 R REAL F.P.
 30 RT REAL F.P.
 0 SIGNX REAL F.P.
 EXTERNALS TYPE ARGS LIBRARY REFERENCES
 COS REAL 1 L,BARY. 9 10
 SIN REAL 1 L,BARY. 9 10

STATISTICS
 PROGRAM LENGTH 5200008 CM USED 31B 25

```

03/13/81 08:26:30 03/13/81 08:26:50
SUBROUTINE READ   73/74  GPT=1          FTN 4.8+508

1      C   SUBROUTINE READ (X,INEW,ANAM,IRD,IOP,T,SR,SD,SH)
C
C   READ IN SUPPLEMENTAL INPUTS
C
5      C
C   DIMENSION X(50),H(9),D(3),V(5),G(3,10),PV(5),ANAM(50),INEW(50)
C   DIMENSION SR(5,2),SD(5,2),SH(5,2)
C   COMMON /RDWT/H,C,V,G,PV,GND,GMH,SDDD(10),SDH(10),IDAT(10)
C   1, RCR(5),BLST(5),HBLST(5),RU,DU
C   DO 11 ID=1,10
C   IDR = IDAT(ID)
C   IF(IIRD.EQ.0)CR.(INEW(IDR).EQ.0.AND.IOPT.EQ.0) GO TO 11
C   IF(X(28).GE.C..AND.ID.EQ.8) GO TO 11
C   NN = 48$X(IDR)
C   IF(NN.EQ.0) GO TO 11
C   IF(IDR.EQ.5) WRITE (6,1009; ANAM(IDR))
C   GO TO (1,2,7,3,4,5,6,8),ID
C   1  IF(X(9)=21,20
C   20 IF(IIRD.EQ.5) WRITE (6,1007) NN
C   READ (IIRD,*); (SDD(I),SDH(I),I=1,NN)
C   GO TO 5
C   21 IF(IIRD.EQ.5) WRITE (6,1008) NN
C   READ (IIRD,*); (SDD(I),I=1,NN)
C   DD 12 I=1,NN
C   SD(I) = SD(I)/1.1774
C   12 SDH(I) = SD(I)
C   9  IF(IIRD.EQ.5) WRITE (6,1012)
C   READ (IIRD,*)
C   GND,GMH
C   GO TO 11
C   2  IF(IIRD.EQ.5) WRITE (6,1005) NN
C   READ (IIRD,*); (SR(I,1),SR(I,2),SD(I,1),SD(I,2),SH(I,1),SH(I,2),I=1,10)
C   C NN,
C   GO TO 11
C   3 NN = NN + 1
C   IF(IIRD.EQ.5) WRITE (6,1000) NN
C   READ (IIRD,*); (H(I),I=1,NN)
C   IF(IIRD.EQ.5) WRITE (6,1014)
C   READ (IIRD,*)
C   RU,DU
C   GO TO 11
C   4 IF(IIRD.EQ.5) WRITE (6,1001) NN
C   READ (IIRD,*); (O(I),I=1,NN)
C   GO TO 11
C   5 IF(IIRD.EQ.5) WRITE (6,1002) NN
C   READ (IIRD,*); (V(I),I=1,NN)
C   IF(IIRD.EQ.5) WRITE (6,1004) NN
C   READ (IIRD,*); (PV(I),I=1,NN)
C   GO TO 11
C   6 IF(IIRD.EQ.5) WRITE (6,1003) NN
C   READ (IIRD,*); ((G(I,J),I=1,3),J=1,NN)
C   GC TC 11
C   7 IF(IIRD.EQ.5) WRITE (6,1005)
C   READ (IIRD,*); (RDR(I),I=1,3)
C   IF(IIRD.EQ.5) WRITE (6,1013)
C   PDR(4),RDR(5)
C   GO TO 11
C   8 IF(IIRD.EQ.5) WRITE (6,1010) NN
C   IF(IIRD.EQ.5) WRITE (6,1011)
C
009950 009960 009980 009990 010000 010010 010020 010030 010040 010050 010060 010070 010080 010090 010100 010110 010120 010130 010140 010150 010160 010170 010180 010190 010200 010210 010220 010230 010240 010250 010260 010270 010280 010290 010300 010310 010320 010330 010340 010350 010360 010370 010380 010390 010400 010410 010420 010430 010440 010450 010460 010470 010480 010490 010500 010510

```

SUBROUTINE READ

73/74

DFT=1

FIN 4.8+503

03/13/81 08.29.30 PAGE 2

```

READ (IRD,*)
 1: CONTINUE
1000 FORMAT (1X,*ENTER *12,* HEIGHTS FOR FRAGMENTATION PK GRID,*,*)
C1X,*LAST VALUE CORRESPONDS TO HEIGHT*,*/
C1X,*WHERE PK GOES TO ZERO*)
1001 FORMAT (1X,*ENTER *12,* ELEVATION ANGLES ASSOC'D TD WITH *)
C /*,10X,*FRAGMENTATION PK DATA -*)
1002 FORMAT (1X,*ENTER *12,* VT FUZING HEIGHTS -*)
1003 FORMAT (1X,*ENTER *12,* SETS OF GLITTER POINT COORDINATES (R,D,H))
C /*,10X,*ENTER *12,* PROB. VT DETONATION AT HEIGHT H - *)
1004 FORMAT (1X,*ENTER *12,* SETS OF BOUNDARIES FOR DIRECT HIT BOXES*)
1005 FORMAT (12,*ENTER *12,* SETS OF BOUNDARIES FOR EACH BOX, MIN RANGE, MAX DEF,*
C /*,*2X HGT, MAX HGT -*)
C /*,* ENTER FOR EACH BOX, MIN RANGE, MAX DEF, MAX DEF,*
1006 FORMAT (1X,*ENTER RADAR ANTENNA COORDINATES (R,D,H) RELATIVE*
C /*,1X,*TG TARGET GROUND ZERO, - *)
1007 FORMAT /*,*ENTER *12,* SETS OF GUIDANCE ERRORS -*,
1/*,3X,*STD DEV DEF, HGT -*,/)
1008 FORMAT (1X,*ENTER *12,* SETS OF GUIDANCE ERRORS -*,
1/*,3X,*CEP -*,/)
1009 FORMAT (1X,A4,1X,*DATA -*)
1010 FORMAT (1X,*ENTER *12,* SETS OF BLST, HGT DATA *)
1011 FORMAT (1X,*BEGINNING WITH LOWEST HEIGHT - *)
1012 FORMAT (1X,*ENTER COORDINATES OF HOMING POINT (R,D,H) - *,/)
1013 FORMAT (1X,*ENTER R1,R2, WHERE RADAR BLAST PK=1*,/)
C1X,*OUT TG R1 AND DECLINES LINEARLY*,*/
C1X,*TO ZERO AT R2 -*)
1014 FORMAT (1X,*ENTER RANGE AND DEFLECTION DISTANCES*/*,
C1X,*FROM EDGE OF GRID TO WHERE THE FRAGMENTATION*,*/
C1X,*PK GOES TO ZERO -*)
END

```

SYMBOLIC REFERENCE MAP (R=2)

ENTRY POINTS DEF LINE REFERENCES

88

VARIABLES	SN	TYPE	RELLOCATION	REFS	16	DEFINED	1
0 ANAM		REAL	ARRAY	RDMRT	REFS	58	
132 BBLST		REAL	ARRAY	RDMRT	REFS	38	
145 DU		REAL	ARRAY	RDMRT	REFS	38	
21 G		REAL	ARRAY	RDMRT	REFS	6	49
65 GMD		REAL	ARRAY	RDMRT	REFS	28	
66 GMH		REAL	ARRAY	RDMRT	REFS	26	
64 GHR		REAL	ARRAY	RDMRT	REFS	28	
0 H		REAL	ARRAY	RDMRT	REFS	6	36
137 HBLST		REAL	ARRAY	RDMRT	REFS	58	
711 I		INTEGER	INTEGER	RDMRT	REFS	2*20	
				RDMRT	REFS	45	2*26
				RDMRT	REFS	31	52
				RDMRT	REFS	41	44
				RDMRT	REFS	36	46
				RDMRT	REFS	58	49
766 ID		INTEGER	ARRAY	RDMRT	REFS	11	36
113 IDAT		INTEGER	ARRAY	RDMRT	REFS	6	20
				RDMRT	REFS	13	36
				RDMRT	REFS	11	49
				RDMRT	REFS	17	49
				RDMRT	REFS	10	49

SUBROUTINE READ			73/74 CPT=1			RELOCATION			FTN 4.6+503			03/13/81 08-29-30			PAGE 3		
VARIABLES	SN	TYPE				REFS	REFS	REFS	2*12	14	16	DEFINED	1	11			
7C7	IDR	INTEGER				F.P.	F.P.	F.P.	12	12	16	DEFINED	1				
0	INEW	INTEGER				REFS	REFS	REFS	16	19	22	DEFINED	1				
0	IOPT	INTEGER				REFS	REFS	REFS	40	43	45	1/G REFS	20	27	30	35	37
0	IRD	INTEGER				DEFINED	DEFINED	DEFINED	1	1/G REFS	48	48	51	53	56	57	57
									38	41	44	46	49	52	54	56	56
712	J	INTEGER				REFS	REFS	REFS	49	49	49	DEFINED	49	52	54	55	58
710	NN	INTEGER				REFS	REFS	REFS	15	19	20	34	35	36	40	41	43
									31	34	36	46	48	49	56	58	58
11	O	REAL				DEFINED	DEFINED	DEFINED	14	34	34	DEFINED	14	41	41		
57	PV	REAL				REFS	REFS	REFS	6	6	8	DEFINED	6	46	46		
125	RDR	REAL				REFS	REFS	REFS	8	8	8	DEFINED	8	52	52		
144	RJ	REAL				REFS	REFS	REFS	8	8	8	DEFINED	8	52	52		
0	SD	REAL				ARRAY	ARRAY	ARRAY	7	7	7	DEFINED	1	2*31	2*31		
67	SSD	REAL				ARRAY	ARRAY	ARRAY	7	7	7	DEFINED	20	26	26	26	26
101	SDH	REAL				ARRAY	ARRAY	ARRAY	8	8	8	DEFINED	20	26	26	26	26
0	SH	REAL				ARRAY	ARRAY	ARRAY	7	7	7	DEFINED	1	2*31	2*31		
0	SR	REAL				ARRAY	ARRAY	ARRAY	7	7	7	DEFINED	1	2*31	2*31		
14	V	REAL				ARRAY	ARRAY	ARRAY	8	8	8	DEFINED	14	44	44		
0	X	REAL				ARRAY	ARRAY	ARRAY	6	13	14	DEFINED	18	44	44		
FILE NAMES						WRITES	WRITES	WRITES	16	19	22	27	30	35	35	37	40
TAPE6									43	45	46	51	53	56	57		
86	VARIABLES USED AS FILE NAMES. SEE ABOVE																
INLINE FUNCTIONS	ARGS	TYPE				1	INTRIN	DEF LINE	REFERENCES	14							
STATEMENT LABELS								DEF LINE	REFERENCES								
46	1							18	17								
117	2							30	17								
145	3							34	17								
165	4							40	17								
200	5							43	17								
223	6							46	17								
236	7							51	17								
253	8							56	17								
110	9							27	21								
277	11							59	16	12	13	15	29	33	39	42	47
0	12							50	55								
0	20									24							
70	21									18							
476	1000	FMT								19							
514	1001	FMT								22	18						
526	1002	FMT								60	35						
534	1003	FMT								63	40						
544	1004	FMT								65	43						
553	1005	FMT								66	49						
574	1006	FMT								68	45						
606	1007	FMT								69	30						
617	1008	FMT								72	51						
627	1009	FMT								74	19						
										76	22						
										76	16						

SUBROUTINE READ			I3/74 OPT=1			FTN 4.8+508			03/13/81 08.29.30			PAGE 4					
STATEMENT LABELS			DEF LINE REFERENCES														
632	1010	FMT		79	55												
640	1011	FMT		80	57												
645	1012	FMT		81	27												
654	1013	FMT		82	53												
670	1014	FMT		85	37												
LOGPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES												
7	I1	ID	10 59	273B	EXT REFS NOT INNER												
56		I	20 26	10B	EXT REFS												
104	I2	I	24 26	3B	INSTACK												
126		I	31 31	15B	EXT REFS												
265		I	58 58	10B	EXT REFS												
COMMON BLOCKS			LENGTH														
	RWRIT	1C2															
STATISTICS																	
PROGRAM LENGTH			747B	487													
CM LABELED COMMAND LENGTH			146B	102													
52000B CM USED																	

SUBROUTINE	WRITE	73/74	OPT=1	FTN 4.8+508	03/13/81	08.29.30	PAGE
1	C	SUBROUTINE WRITE (X,IWRT,CEP,SR,SD,SH)					1
5	C	WRITE LIST OF DATA (OUTPUT & TAPE1)					
10	C	DIMENSION X(50),H(5),Q(3),V(5),G(3,10),PV(5),CEP(10)			010830		
10	C	COMMON SD(SR(5,2),SH(5,2))			010840		
10	C	CYTHON /ROKAT/ H,G,V,G,PV,GMR,GMD,GMH,SDD(10),SDH(10),IDAT(10)			010850		
10	C	1,RDR(5),BBLST(5),HBLST(5),RU,DU			010860		
10	C	DO 8 I=1,50			010880		
10	C	NN = ABS(X(I))			010890		
10	C	IF(NN.EQ.0) GO TO 8			010900		
10	C	DO 10 J=1,10			010910		
10	C	JJ = J			010920		
10	C	IF(IDAT(JJ).EQ.1) GO TO 11			010930		
10	C	GO TO 6			010940		
11	C	GO TO (1,2,7,3,4,5,6,9),JJ			010950		
11	C	: IF(X(19)) 21,20			010960		
20	C	20 WRITE (IWRT,*) (SDD(K),SDH(K),K=1,NN)			010970		
20	C	GO TO 13			010980		
21	C	21 DO 12 K=1,NN			010990		
21	C	12 CEP(K) = SDD(K)*1.1774			011000		
21	C	WRITE (IWRT,*) (CEP(K),K=1,NN)			011010		
21	C	13 WRITE (IWRT,*) GMR,GMD,GMH			011020		
21	C	GO TO 8			011030		
21	C	2 WRITE (IWRT,*) (SR(K,1),SR(K,2),SD(K,1),SD(K,2),SH(K,1),SH(K,2),			011040		
21	C	1, K=1,NN)			011050		
21	C	GO TO 8			011060		
30	C	3 NN = NN + 1			011070		
30	C	4 WRITE (IWRT,*) (H(K),K=1,NN)			011080		
30	C	5 WRITE (IWRT,*) RU,CU			011090		
30	C	6 GO TO 8			011100		
35	C	7 GO TO 8			011110		
35	C	8 GO TO 8			011120		
35	C	9 GO TO 8			011130		
35	C	10 GO TO 8			011140		
35	C	11 GO TO 8			011150		
35	C	12 GO TO 8			011160		
35	C	13 GO TO 8			011170		
35	C	14 GO TO 8			011180		
35	C	15 GO TO 8			011190		
40	C	16 GO TO 8			011200		
40	C	17 GO TO 8			011210		
40	C	18 GO TO 8			011220		
40	C	19 GO TO 8			011230		
45	C	20 GO TO 8			011240		
45	C	21 GO TO 8			011250		
45	C	22 GO TO 8			011260		
45	C	23 GO TO 8			011270		
45	C	24 GO TO 8			011280		

REFERRAL
47

SUBROUTINE WRITE		73/74	OPT=1		FTN 4.8+508	03/13/81	08.29.30	PAGE	2
VARIABLES		SN	TYPE	RELOCATION					
132 BSLST		REAL	ARRAY	R2WRT	REFS	8	45	DEFINED	1
0 CEP		REAL	ARRAY	F.P.	REFS	6	24	DEFINED	1
145 DU		REAL	ARRAY	R2WRT	REFS	6	32		23
21 G		REAL	ARRAY	R2WRT	REFS	6	6		39
65 GM2		REAL	ARRAY	R2WRT	REFS	6	25		
66 GM3		REAL	ARRAY	R2WRT	REFS	6	25		
64 GM2		REAL	ARRAY	R2WRT	REFS	6	25		
0 H		REAL	ARRAY	R2WRT	REFS	6	8		
137 HBLST		REAL	ARRAY	R2WRT	REFS	6	45		
276 I		INTEGER	ARRAY	R2WRT	REFS	11	15	DEFINED	16
113 IDAT		INTEGER	ARRAY	R2WRT	REFS	8	15	DEFINED	16
0 INRT		INTEGER	ARRAY	F.P.	DEFINED	1	1/O REFS	20	25
300 J		INTEGER	ARRAY	R2WRT	REFS	32	36	37	25
301 JJ		INTEGER	ARRAY	R2WRT	REFS	14	41	39	41
352 K		INTEGER	ARRAY	R2WRT	REFS	15	18	DEFINED	42
303 L		INTEGER	ARRAY	R2WRT	REFS	2*20	2*23	14	45
277 NN		INTEGER	ARRAY	R2WRT	REFS	37	39	24	45
11 Q		REAL	ARRAY	R2WRT	REFS	34	36	2*45	45
57 PV		REAL	ARRAY	R2WRT	REFS	12	20	37	45
125 RDR		REAL	ARRAY	R2WRT	REFS	6	6	39	45
142 RU		REAL	ARRAY	R2WRT	REFS	6	8	37	45
66 SD0		REAL	ARRAY	F.P.	REFS	6	41	2*42	
67 SD1		REAL	ARRAY	R2WRT	REFS	6	32		
101 SDH		REAL	ARRAY	R2WRT	REFS	7	2*27	DEFINED	1
0 SH		REAL	ARRAY	R2WRT	REFS	8	20	23	
0 SR		REAL	ARRAY	R2WRT	REFS	7	2*27	DEFINED	1
14 V		REAL	ARRAY	R2WRT	REFS	6	8	36	
C X		REAL	ARRAY	F.P.	REFS	6	11	19	44
VARIABLES USED AS FILE NAMES, SEE ABOVE								DEFINED	1
INLINE FUNCTIONS		TYPE	ARGS	INTRIN	DEF LINE	REFERENCES			
ABS		REAL	1		11				
STATEMENT LABELS				GEF LINE	REFERENCES				
27 1				19	18				
74 2				27	18				
116 3				30	18				
127 4				34	18				
135 5				36	18				
156 6				39	18				
157 7				41	18				
202 6				46	10	12	17	26	38
164 8				44	18				40
0 10				16	13				
22 11				18	15				
0 12				23	22				
71 13				25	21				
0 20				20	19				
55 21				22	19				
71 13				INACTIVE					

PAGE 3

03/13/81 08.29.30

FTN 4.8+508

SUBROUTINE WRITE 73/74 OPT=1

LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES
7	8	I	10 46	176B	EXT REFS
14	16	J	13 16	53	INSTACK EXITS
43		K	20 20	108	EXT REFS
60	12	K	22 23	38	INSTACK
77		K	27 27	158	EXT REFS
171		J	45 45	108	EXT REFS

COMMON BLOCKS	LENGTH
RDWT	102

STATISTICS

PROGRAM LENGTH	320B	208
CM LABELED COMMON LENGTH	1468	102
5200GB CM USED		

SUBROUTINE GRIDS 73/74 QFT=1 FTN 4.8+508 03/13/81 08.29.30 PAGE 1

```

1      C   SUBROUTINE GRIDS (PK,NH,KK,R,D,NR,ND,NDBG)
C   READ IN FRAGMENTATION PK GRID REDEFINE AND ORIENT
C   AXES TO CORRESPOND WITH GEOMETRY OF MODEL
5      C   GRIDS ARE IN ROTATED PROJECTILE COORDINATE SYSTEM.
C
C   DIMENSION PK(40,20,8),R(8,41),D(8,21)
DC 1  I=1,NH
      READ (KK,1001) NR,ND
      IF (NCBG.EQ.5) WRITE (6,2001) NR,ND
      READ (KK,1000) (R(I,NR-J+1),J=1,NR)
      READ (KK,1000) (D(I,ND-J+1),J=1,ND)
      NR = NR-1
      ND = ND-1
C
C   REDDEFINE GRIDS AT CENTER OF CELLS (AT PK)
C   AND CHANGE SIGN OF GRID COORDINATES AND
C   CHANGE ALL INDICES TO GET GRID COORDINATES
C   IN ASCENDING ORDER AND IN PROPER RELATIONSHIP
C   TG ARPSIM GEOMETRY.
C
C   DC 4 J=1,NR
        4 R(I,J) = -(R(I,J) + R(I,J+1))/2.
C   DC 5 J=1,ND
        5 D(I,J) = -(D(I,J) + D(I,J+1))/2.
        IF (NDBG.EQ.5) WRITE (6,2000) (R(I,J),J=1,NR)
        IF (NDBG.EQ.5) WRITE (6,2000) (D(I,J),J=1,ND)
        DG 1 J=1,NR
        1 READ (KK,1002) (PK(NR-J+1,ND-K+1,I),K=1,ND)
        IF (NDBG.NE.5) RETURN
        DC 2 I=1,NH
        DC 2 J=1,NR
        WRITE (6,2002) (PK(J,K,I),K=1,ND)
C
C   CONTINUE
        2 CONTINUE
        1000 FORMAT (10F7.1)
        1001 FORMAT (2I3)
        1002 FORMAT (10F7.5)
        2000 FORMAT (1X,1CF7.1)
        2001 FORMAT (1X,2I3)
        2002 FORMAT (1X,10F7.5)
        END
  
```

91

SYMBOLIC REFERENCE MAP (R=2)

ENTRY POINTS	DEF LINE	REFERENCES				
3 GRIDS	1	32	43			
VARIABLES	SN TYPE	RELOCATION				
0 0	REAL	ARRAY				
307 1	INTEGER	F.P.				
	REFS	9	2*27	29	DEFINED	1
	REFS	13	14	3*25	3*27	28
	REFS	35	DEFINED	10	35	29
						31
						27
						14
						29

SUBROUTINE GRIDS			73/74	OFT=1				
VARIABLES	SN	TYPE	RELOCATION					
210 J		INTEGER	F.P.	DEFINED	REFS	13	14	3*25
	0 KK	INTEGER	F.P.	DEFINED	REFS	13	14	3*27
	C ND	INTEGER	F.P.	DEFINED	REFS	31	11	24
311 K		INTEGER	F.P.	DEFINED	REFS	35	11	26
0 NH		INTEGER	F.P.	DEFINED	REFS	31	13	28
C NR		INTEGER	F.P.	DEFINED	REFS	31	14	29
C NDBG		INTEGER	F.P.	DEFINED	REFS	12	16	29
0 NH		INTEGER	F.P.	DEFINED	REFS	10	16	29
C NR		INTEGER	F.P.	DEFINED	REFS	12	16	29
C PK		REAL	ARRAY	DEFINED	REFS	1	28	31
C R		REAL	ARRAY	DEFINED	REFS	1	24	31
FILE NAMES		MODE	F.P.	DEFINED	REFS	1	24	31
TAPE6		FMT	F.P.	DEFINED	REFS	1	15	31
VARIABLES USED AS FILE NAMES, SEE ABOVE						1	11	31
						12	28	31
						28	29	35

FILE NAMES
TAPE6
VARIABLES USED AS FILE NAMES, SEE ABOVE

STATEMENT	LABELS		DEF LINE	REFERENCES				
6 1			31	10	30			
6 2			36	33	34			
0 4			25	24				
0 5			27	26				
271 1000	FMT		37	13	14			
273 1001	FMT		38	11				
275 1002	FMT		39	31				
277 2000	FMT		40	28	29			
301 2001	FMT		41	12				
303 2002	FMT		42	35				
LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES	EXT REFS	NOT INNER	
7 1	I		10 31	150B		EXT REFS	NOT INNER	
20	J		13 13	11B		EXT REFS	NOT INNER	
35	J		14 14	11B		EXT REFS	NOT INNER	
56 4	J		24 25	4B	INSTACK	EXT REFS	NOT INNER	
70 5	J		26 27	4B	INSTACK	EXT REFS	NOT INNER	
101	J		28 28	10B		EXT REFS	NOT INNER	
117	J		29 29	10B		EXT REFS	NOT INNER	
131 1	J K		30 31	24B		EXT REFS	NOT INNER	
134	K		31 31	15B		EXT REFS	NOT INNER	
162 2	I		33 36	24B		EXT REFS	NOT INNER	
163 2	J		34 36	21B		EXT REFS	NOT INNER	
166	K		35 35	12B		EXT REFS	NOT INNER	

STATISTICS
PROGRAM LENGTH 350B
52000B CM USED 232

SUBROUTINE SEARCH 73/74 OPT=1

FTN 4.8+506 03/13/81 08.29.30

PAGE 1

```

1      C
      C   SUBROUTINE SEARCH ( I1,J1,R,D,H )
      C
      C   DETERMINES WHETHER INTERCEPT OF TRAJECTORY WITH DIRECT
      C   HIT BOX PLANES FALLS WITHIN BOUNDARY OF DIRECT HIT BOX.
      C   UPDATES PENETRATION (BOX INTERCEPT) COORDINATES (RPN,DPN,HPN) 011750
      C   IF HEIGHT COMPONENT INDICATES BOX PENETRATION OCCURS 011760
      C   PRIOR TO CURRENT HPN POINT ALONG TRAJECTORY. INITIAL VALUE 011770
      C   OF HPN IS EASED ON BURST POINT HEIGHT. BH. 011780
      C
      C   COMMON /SRCH/ IPEN,IPN,RBS,DBS,MBS,BR,BD,BH,OMEGA,RPN,DPN
      C   C,HPN,RDH1,RDH2,DDH1,DDH2,HDH1,HDH2 011790
      C   DX((R1,R2,D1,D2,H1,H2)) = (R2-R1)*(R2-D1)+(D2-D1)*(H2-H1)*(H2-H1) 011800
      C   C2-H1) 011810
      C   CALL INTRCP (R,D,H,UJ)
      C   IF(R.GT.RDH2.OR.R.LT.RDH1) RETURN 011820
      C   IF(D.GT.DDH2.OR.D.LT.DDH1) RETURN 011830
      C   IF(H.GT.HDH2.OR.H.LT.HDH1) RETURN 011840
      C   IPEN = IPEN + 1 011850
      C   IF(OMEGA.GE.0..AND.H.LT.HPN) RETURN 011860
      C   IF(OMEGA.LT.0..AND.H.GT.HPN) RETURN 011870
      C   RPN = R 011880
      C   DPN = D 011890
      C   HPN = H 011900
      C   IPN = 11 011910
      C   END 011920
      C
      C   15
      C   20
      C   25
      C
      C   SYMBOLIC REFERENCE MAP (R=2)
      C
      C   ENTRY POINTS    DEF LINE    REFERENCES    RELOCATION
      C   3    SEARCH    1    15    16    17    19    20    25
      C
      C   VARIABLES    SN    TYPE
      C   6    BD    REAL
      C   7    BH    REAL
      C   5    BR    REAL
      C   0    D    REAL
      C   3    DBS    REAL
      C   16    DDH1    REAL
      C   17    DDH2    REAL
      C   12    DPN    REAL
      C   0    H    REAL
      C   4    MBS    REAL
      C   20    HDH1    REAL
      C   21    HDH2    REAL
      C   13    HPN    REAL
      C   0    I1    INTEGER
      C   0    IPEN    INTEGER
      C   1    IPN    INTEGER
      C   2    UJ    INTEGER
      C   10    OMEGA    REAL
      C   0    R    REAL
      C   2    RBS    REAL
      C
      C   1
      C
      C   15    SRCH    REFS    1C
      C   16    SRCH    REFS    10
      C   17    SRCH    REFS    10
      C   18    F.P.    REFS    14
      C   19    SRCH    REFS    10
      C   20    SRCH    REFS    10
      C   21    SRCH    REFS    10
      C   22    F.P.    REFS    10
      C   23    SRCH    REFS    10
      C   24    SRCH    REFS    10
      C   25    SRCH    REFS    10
      C   26    F.P.    REFS    14
      C   27    SRCH    REFS    10
      C   28    F.P.    REFS    10
      C   29    SRCH    REFS    10
      C   30    F.P.    REFS    14
      C   31    SRCH    REFS    10
      C   32    F.P.    REFS    10
      C   33    SRCH    REFS    10
      C   34    F.P.    REFS    14
      C   35    SRCH    REFS    10
      C   36    F.P.    REFS    10
      C   37    SRCH    REFS    10
      C   38    F.P.    REFS    14
      C   39    SRCH    REFS    10
      C   40    F.P.    REFS    10
      C   41    SRCH    REFS    10
      C   42    F.P.    REFS    14
      C   43    SRCH    REFS    10
      C   44    F.P.    REFS    10
      C   45    SRCH    REFS    10
      C   46    F.P.    REFS    14
      C   47    SRCH    REFS    10
      C   48    F.P.    REFS    10
      C   49    SRCH    REFS    10
      C   50    F.P.    REFS    14
      C   51    SRCH    REFS    10
      C   52    F.P.    REFS    10
      C   53    SRCH    REFS    10
      C   54    F.P.    REFS    14
      C   55    SRCH    REFS    10
      C   56    F.P.    REFS    10
      C   57    SRCH    REFS    10
      C   58    F.P.    REFS    14
      C   59    SRCH    REFS    10
      C   60    F.P.    REFS    10
      C   61    SRCH    REFS    10
      C   62    F.P.    REFS    14
      C   63    SRCH    REFS    10
      C   64    F.P.    REFS    10
      C   65    SRCH    REFS    10
      C   66    F.P.    REFS    14
      C   67    SRCH    REFS    10
      C   68    F.P.    REFS    10
      C   69    SRCH    REFS    10
      C   70    F.P.    REFS    14
      C   71    SRCH    REFS    10
      C   72    F.P.    REFS    10
      C   73    SRCH    REFS    10
      C   74    F.P.    REFS    14
      C   75    SRCH    REFS    10
      C   76    F.P.    REFS    10
      C   77    SRCH    REFS    10
      C   78    F.P.    REFS    14
      C   79    SRCH    REFS    10
      C   80    F.P.    REFS    10
      C   81    SRCH    REFS    10
      C   82    F.P.    REFS    14
      C   83    SRCH    REFS    10
      C   84    F.P.    REFS    10
      C   85    SRCH    REFS    10
      C   86    F.P.    REFS    14
      C   87    SRCH    REFS    10
      C   88    F.P.    REFS    10
      C   89    SRCH    REFS    10
      C   90    F.P.    REFS    14
      C   91    SRCH    REFS    10
      C   92    F.P.    REFS    10
      C   93    SRCH    REFS    10
      C   94    F.P.    REFS    14
      C   95    SRCH    REFS    10
      C   96    F.P.    REFS    10
      C   97    SRCH    REFS    10
      C   98    F.P.    REFS    14
      C   99    SRCH    REFS    10
      C   100    F.P.    REFS    10
      C   101    SRCH    REFS    10
      C   102    F.P.    REFS    14
      C   103    SRCH    REFS    10
      C   104    F.P.    REFS    10
      C   105    SRCH    REFS    10
      C   106    F.P.    REFS    14
      C   107    SRCH    REFS    10
      C   108    F.P.    REFS    10
      C   109    SRCH    REFS    10
      C   110    F.P.    REFS    14
      C   111    SRCH    REFS    10
      C   112    F.P.    REFS    10
      C   113    SRCH    REFS    10
      C   114    F.P.    REFS    14
      C   115    SRCH    REFS    10
      C   116    F.P.    REFS    10
      C   117    SRCH    REFS    10
      C   118    F.P.    REFS    14
      C   119    SRCH    REFS    10
      C   120    F.P.    REFS    10
      C   121    SRCH    REFS    10
      C   122    F.P.    REFS    14
      C   123    SRCH    REFS    10
      C   124    F.P.    REFS    10
      C   125    SRCH    REFS    10
      C   126    F.P.    REFS    14
      C   127    SRCH    REFS    10
      C   128    F.P.    REFS    10
      C   129    SRCH    REFS    10
      C   130    F.P.    REFS    14
      C   131    SRCH    REFS    10
      C   132    F.P.    REFS    10
      C   133    SRCH    REFS    10
      C   134    F.P.    REFS    14
      C   135    SRCH    REFS    10
      C   136    F.P.    REFS    10
      C   137    SRCH    REFS    10
      C   138    F.P.    REFS    14
      C   139    SRCH    REFS    10
      C   140    F.P.    REFS    10
      C   141    SRCH    REFS    10
      C   142    F.P.    REFS    14
      C   143    SRCH    REFS    10
      C   144    F.P.    REFS    10
      C   145    SRCH    REFS    10
      C   146    F.P.    REFS    14
      C   147    SRCH    REFS    10
      C   148    F.P.    REFS    10
      C   149    SRCH    REFS    10
      C   150    F.P.    REFS    14
      C   151    SRCH    REFS    10
      C   152    F.P.    REFS    10
      C   153    SRCH    REFS    10
      C   154    F.P.    REFS    14
      C   155    SRCH    REFS    10
      C   156    F.P.    REFS    10
      C   157    SRCH    REFS    10
      C   158    F.P.    REFS    14
      C   159    SRCH    REFS    10
      C   160    F.P.    REFS    10
      C   161    SRCH    REFS    10
      C   162    F.P.    REFS    14
      C   163    SRCH    REFS    10
      C   164    F.P.    REFS    10
      C   165    SRCH    REFS    10
      C   166    F.P.    REFS    14
      C   167    SRCH    REFS    10
      C   168    F.P.    REFS    10
      C   169    SRCH    REFS    10
      C   170    F.P.    REFS    14
      C   171    SRCH    REFS    10
      C   172    F.P.    REFS    10
      C   173    SRCH    REFS    10
      C   174    F.P.    REFS    14
      C   175    SRCH    REFS    10
      C   176    F.P.    REFS    10
      C   177    SRCH    REFS    10
      C   178    F.P.    REFS    14
      C   179    SRCH    REFS    10
      C   180    F.P.    REFS    10
      C   181    SRCH    REFS    10
      C   182    F.P.    REFS    14
      C   183    SRCH    REFS    10
      C   184    F.P.    REFS    10
      C   185    SRCH    REFS    10
      C   186    F.P.    REFS    14
      C   187    SRCH    REFS    10
      C   188    F.P.    REFS    10
      C   189    SRCH    REFS    10
      C   190    F.P.    REFS    14
      C   191    SRCH    REFS    10
      C   192    F.P.    REFS    10
      C   193    SRCH    REFS    10
      C   194    F.P.    REFS    14
      C   195    SRCH    REFS    10
      C   196    F.P.    REFS    10
      C   197    SRCH    REFS    10
      C   198    F.P.    REFS    14
      C   199    SRCH    REFS    10
      C   200    F.P.    REFS    10
      C   201    SRCH    REFS    10
      C   202    F.P.    REFS    14
      C   203    SRCH    REFS    10
      C   204    F.P.    REFS    10
      C   205    SRCH    REFS    10
      C   206    F.P.    REFS    14
      C   207    SRCH    REFS    10
      C   208    F.P.    REFS    10
      C   209    SRCH    REFS    10
      C   210    F.P.    REFS    14
      C   211    SRCH    REFS    10
      C   212    F.P.    REFS    10
      C   213    SRCH    REFS    10
      C   214    F.P.    REFS    14
      C   215    SRCH    REFS    10
      C   216    F.P.    REFS    10
      C   217    SRCH    REFS    10
      C   218    F.P.    REFS    14
      C   219    SRCH    REFS    10
      C   220    F.P.    REFS    10
      C   221    SRCH    REFS    10
      C   222    F.P.    REFS    14
      C   223    SRCH    REFS    10
      C   224    F.P.    REFS    10
      C   225    SRCH    REFS    10
      C   226    F.P.    REFS    14
      C   227    SRCH    REFS    10
      C   228    F.P.    REFS    10
      C   229    SRCH    REFS    10
      C   230    F.P.    REFS    14
      C   231    SRCH    REFS    10
      C   232    F.P.    REFS    10
      C   233    SRCH    REFS    10
      C   234    F.P.    REFS    14
      C   235    SRCH    REFS    10
      C   236    F.P.    REFS    10
      C   237    SRCH    REFS    10
      C   238    F.P.    REFS    14
      C   239    SRCH    REFS    10
      C   240    F.P.    REFS    10
      C   241    SRCH    REFS    10
      C   242    F.P.    REFS    14
      C   243    SRCH    REFS    10
      C   244    F.P.    REFS    10
      C   245    SRCH    REFS    10
      C   246    F.P.    REFS    14
      C   247    SRCH    REFS    10
      C   248    F.P.    REFS    10
      C   249    SRCH    REFS    10
      C   250    F.P.    REFS    14
      C   251    SRCH    REFS    10
      C   252    F.P.    REFS    10
      C   253    SRCH    REFS    10
      C   254    F.P.    REFS    14
      C   255    SRCH    REFS    10
      C   256    F.P.    REFS    10
      C   257    SRCH    REFS    10
      C   258    F.P.    REFS    14
      C   259    SRCH    REFS    10
      C   260    F.P.    REFS    10
      C   261    SRCH    REFS    10
      C   262    F.P.    REFS    14
      C   263    SRCH    REFS    10
      C   264    F.P.    REFS    10
      C   265    SRCH    REFS    10
      C   266    F.P.    REFS    14
      C   267    SRCH    REFS    10
      C   268    F.P.    REFS    10
      C   269    SRCH    REFS    10
      C   270    F.P.    REFS    14
      C   271    SRCH    REFS    10
      C   272    F.P.    REFS    10
      C   273    SRCH    REFS    10
      C   274    F.P.    REFS    14
      C   275    SRCH    REFS    10
      C   276    F.P.    REFS    10
      C   277    SRCH    REFS    10
      C   278    F.P.    REFS    14
      C   279    SRCH    REFS    10
      C   280    F.P.    REFS    10
      C   281    SRCH    REFS    10
      C   282    F.P.    REFS    14
      C   283    SRCH    REFS    10
      C   284    F.P.    REFS    10
      C   285    SRCH    REFS    10
      C   286    F.P.    REFS    14
      C   287    SRCH    REFS    10
      C   288    F.P.    REFS    10
      C   289    SRCH    REFS    10
      C   290    F.P.    REFS    14
      C   291    SRCH    REFS    10
      C   292    F.P.    REFS    10
      C   293    SRCH    REFS    10
      C   294    F.P.    REFS    14
      C   295    SRCH    REFS    10
      C   296    F.P.    REFS    10
      C   297    SRCH    REFS    10
      C   298    F.P.    REFS    14
      C   299    SRCH    REFS    10
      C   300    F.P.    REFS    10
      C   301    SRCH    REFS    10
      C   302    F.P.    REFS    14
      C   303    SRCH    REFS    10
      C   304    F.P.    REFS    10
      C   305    SRCH    REFS    10
      C   306    F.P.    REFS    14
      C   307    SRCH    REFS    10
      C   308    F.P.    REFS    10
      C   309    SRCH    REFS    10
      C   310    F.P.    REFS    14
      C   311    SRCH    REFS    10
      C   312    F.P.    REFS    10
      C   313    SRCH    REFS    10
      C   314    F.P.    REFS    14
      C   315    SRCH    REFS    10
      C   316    F.P.    REFS    10
      C   317    SRCH    REFS    10
      C   318    F.P.    REFS    14
      C   319    SRCH    REFS    10
      C   320    F.P.    REFS    10
      C   321    SRCH    REFS    10
      C   322    F.P.    REFS    14
      C   323    SRCH    REFS    10
      C   324    F.P.    REFS    10
      C   325    SRCH    REFS    10
      C   326    F.P.    REFS    14
      C   327    SRCH    REFS    10
      C   328    F.P.    REFS    10
      C   329    SRCH    REFS    10
      C   330    F.P.    REFS    14
      C   331    SRCH    REFS    10
      C   332    F.P.    REFS    10
      C   333    SRCH    REFS    10
      C   334    F.P.    REFS    14
      C   335    SRCH    REFS    10
      C   336    F.P.    REFS    10
      C   337    SRCH    REFS    10
      C   338    F.P.    REFS    14
      C   339    SRCH    REFS    10
      C   340    F.P.    REFS    10
      C   341    SRCH    REFS    10
      C   342    F.P.    REFS    14
      C   343    SRCH    REFS    10
      C   344    F.P.    REFS    10
      C   345    SRCH    REFS    10
      C   346    F.P.    REFS    14
      C   347    SRCH    REFS    10
      C   348    F.P.    REFS    10
      C   349    SRCH    REFS    10
      C   350    F.P.    REFS    14
      C   351    SRCH    REFS    10
      C   352    F.P.    REFS    10
      C   353    SRCH    REFS    10
      C   354    F.P.    REFS    14
      C   355    SRCH    REFS    10
      C   356    F.P.    REFS    10
      C   357    SRCH    REFS    10
      C   358    F.P.    REFS    14
      C   359    SRCH    REFS    10
      C   360    F.P.    REFS    10
      C   361    SRCH    REFS    10
      C   362    F.P.    REFS    14
      C   363    SRCH    REFS    10
      C   364    F.P.    REFS    10
      C   365    SRCH    REFS    10
      C   366    F.P.    REFS    14
      C   367    SRCH    REFS    10
      C   368    F.P.    REFS    10
      C   369    SRCH    REFS    10
      C   370    F.P.    REFS    14
      C   371    SRCH    REFS    10
      C   372    F.P.    REFS    10
      C   373    SRCH    REFS    10
      C   374    F.P.    REFS    14
      C   375    SRCH    REFS    10
      C   376    F.P.    REFS    10
      C   377    SRCH    REFS    10
      C   378    F.P.    REFS    14
      C   379    SRCH    REFS    10
      C   380    F.P.    REFS    10
      C   381    SRCH    REFS    10
      C   382    F.P.    REFS    14
      C   383    SRCH    REFS    10
      C   384    F.P.    REFS    10
      C   385    SRCH    REFS    10
      C   386    F.P.    REFS    14
      C   387    SRCH    REFS    10
      C   388    F.P.    REFS    10
      C   389    SRCH    REFS    10
      C   390    F.P.    REFS    14
      C   391    SRCH    REFS    10
      C   392    F.P.    REFS    10
      C   393    SRCH    REFS    10
      C   394    F.P.    REFS    14
      C   395    SRCH    REFS    10
      C   396    F.P.    REFS    10
      C   397    SRCH    REFS    10
      C   398    F.P.    REFS    14
      C   399    SRCH    REFS    10
      C   400    F.P.    REFS    10
      C   401    SRCH    REFS    10
      C   402    F.P.    REFS    14
      C   403    SRCH    REFS    10
      C   404    F.P.    REFS    10
      C   405    SRCH    REFS    10
      C   406    F.P.    REFS    14
      C   407    SRCH    REFS    10
      C   408    F.P.    REFS    10
      C   409    SRCH    REFS    10
      C   410    F.P.    REFS    14
      C   411    SRCH    REFS    10
      C   412    F.P.    REFS    10
      C   413    SRCH    REFS    10
      C   414    F.P.    REFS    14
      C   415    SRCH    REFS    10
      C   416    F.P.    REFS    10
      C   417    SRCH    REFS    10
      C   418    F.P.    REFS    14
      C   419    SRCH    REFS    10
      C   420    F.P.    REFS    10
      C   421    SRCH    REFS    10
      C   422    F.P.    REFS    14
      C   423    SRCH    REFS    10
      C   424    F.P.    REFS    10
      C   425    SRCH    REFS    10
      C   426    F.P.    REFS    14
      C   427    SRCH    REFS    10
      C   428    F.P.    REFS    10
      C   429    SRCH    REFS    10
      C   430    F.P.    REFS    14
      C   431    SRCH    REFS    10
      C   432    F.P.    REFS    10
      C   433    SRCH    REFS    10
      C   434    F.P.    REFS    14
      C   435    SRCH    REFS    10
      C   436    F.P.    REFS    10
      C   437    SRCH    REFS    10
      C   438    F.P.    REFS    14
      C   439    SRCH    REFS    10
      C   440    F.P.    REFS    10
      C   441    SRCH    REFS    10
      C   442    F.P.    REFS    14
      C   443    SRCH    REFS    10
      C   444    F.P.    REFS    10
      C   445    SRCH    REFS    10
      C   446    F.P.    REFS    14
      C   447    SRCH    REFS    10
      C   448    F.P.    REFS    10
      C   449    SRCH    REFS    10
      C   450    F.P.    REFS    14
      C   451    SRCH    REFS    10
      C   452    F.P.    REFS    10
      C   453    SRCH    REFS    10
      C   454    F.P.    REFS    14
      C   455    SRCH    REFS    10
      C   456    F.P.    REFS    10
      C   457    SRCH    REFS    10
      C   458    F.P.    REFS    14
      C   459    SRCH    REFS    10
      C   460    F.P.    REFS    10
      C   461    SRCH    REFS    10
      C   462    F.P.    REFS    14
      C   463    SRCH    REFS    10
      C   464    F.P.    REFS    10
      C   465    SRCH    REFS    10
      C   466    F.P.    REFS    14
      C   467    SRCH    REFS    10
      C   468    F.P.    REFS    10
      C   469    SRCH    REFS    10
      C   470    F.P.    REFS    14
      C   471    SRCH    REFS    10
      C   472    F.P.    REFS    10
      C   473    SRCH    REFS    10
      C   474    F.P.    REFS    14
      C   475    SRCH    REFS    10
      C   476    F.P.    REFS    10
      C   477    SRCH    REFS    10
      C   478    F.P.    REFS    14
      C   479    SRCH    REFS    1
```

VARIABLES

SN

TYPE

REFS

PAGE

14 RDH1 REAL

10

2

15 RDH2 REAL

15

15

11 RPN REAL

10

DEFINED

21

EXTERNALS

INTRCP

TYPE

REFS

REFERENCES

INLINE FUNCTIONS

DX

REAL

DEF

14

6

LINE

REFERENCES

COMMON BLOCKS

SRCH

LENGTH

SF

12

STATISTICS

PROGRAM LENGTH

18

18

CM LABELED COMMON LENGTH

466

38

52000B CM USED

226

18

PAGE 1

FTN 4.8+SOB 03/13/81 08.29.30

SUBROUTINE INTRCP 73/74 CPT=1

```

4      C          SUBROUTINE INTRCP (R,D,H,IGO)
C          COMPUTES INTERCEPT OF TRAJECTORY WITH SIDES OF TARGET
C          BOXES USING TWO POINTS. (BR,BD,BH) AND (RBS,DBS,HBS).
C          INTERCEPT IS AT (R,D,H). AND (RBS,DBS,HBS).
C          SEE MAIN ROUTINE BETWEEN STATEMENTS 105 AND 96.
C
C          COMMON /SRCH/ IPEN,IBS,RBS,DBS,HBS,BR,BD,BH,OMEGA,RPN,DPN
C,HPN,RDH1,RDH2,DDH1,DDH2,HDH1,HDH2
C,XOFY(GX,XA,Y,GY,YA)=GX+(XA-GX)*(YA-GY)/(YA-GY)
C          GO TO (1,2,3),IGO
C
C          GIVEN R, SOLVE FOR D,H
C
15     C          1 D = XOFY(BD,DBS,R,BR,RBS)
C          H = XOFY(BH,HBS,R,BR,RBS)
C          RETURN
C
C          GIVEN D, SOLVE FOR R,H
C
20     C          2 R = XOFY(BR,RBS,D,BD,DBS)
C          H = XOFY(BH,HBS,D,BD,DBS)
C          RETURN
C
C          GIVEN H, SOLVE FOR R,D
C
25     C          3 R = XOFY(BR,RBS,H,BH,HBS)
C          D = XOFY(BD,DBS,H,BH,HBS)
C          RETURN
C          END
C
30

```

CARD NR. SEVERITY DETAILS DIAGNOSIS OF PROBLEM

11 1 AN IF STATEMENT MAY BE MORE EFFICIENT THAN A 2 OR 3 BRANCH COMPUTED GO TO STATEMENT.

SYMBOLIC REFERENCE MAP (R=x2)

ENTRY POINTS	DEF LINE	REFERENCES	RELOCATION	REFERENCES	2*21	2*22	2*28
3 INTRCP	1	17	SRCH	REFS	8	2*15	2*21
6 BD	REAL	SRCH	REFS	8	2*16	2*22	2*28
7 BH	REAL	SRCH	REFS	8	2*15	2*16	2*27
5 BR	REAL	F.P.	REFS	21	22	DEFINED	1 15 28
0 D	REAL	SRCH	REFS	8	15	21	22
3 DBS	REAL	SRCH	REFS	8			
16 DDH1	REAL	SRCH	REFS	8			
17 DDH2	REAL	SRCH	REFS	8			
12 DPN	REAL	SRCH	REFS	8			
C H	REAL	F.P.	REFS	27	28	DEFINED	1 1E 22

SUBROUTINE INTRCP		73/74	OPT*1	FTN 4.8+508			03/13/81	08.25.30	PAGE
VARIABLES	SN	TYPE	RELOCATION						2
4 HBS		REAL	SRCH	REFS	8		16	22	27
20 HDH1		REAL	SRCH	REFS	8				28
21 HDH2		REAL	SRCH	REFS	8				
13 HPN		REAL	SRCH	REFS	8				
1 IBS		INTEGER	SRCH	REFS	8				
0 IGO		INTEGER	F.P.	REFS	11	DEFINED	1		
0 IPEN		INTEGER	SRCH	REFS	8				
10 OMEGA		REAL	SRCH	REFS	8				
0 R		REAL	F.P.	REFS	15				
2 RBS		REAL	SRCH	REFS	8				
14 RDH1		REAL	SRCH	REFS	8				
15 RDH2		REAL	SRCH	REFS	8				
11 RPN		REAL	SRCH	REFS	8				
INLINE FUNCTIONS		TYPE	ARGS	DEF LINE	REFERENCES				
XQFY		REAL	5 SF	10	15	16	21	22	27
STATEMENT LABELS			DEF LINE	REFERENCES					
15 1			15	11					
27 2			21	11					
41 3			27	11					
COMMON BLOCKS		LENGTH							
SRCH		18							
STATISTICS									
PROGRAM LENGTH									
CM LABELED COMMON LENGTH			53B	43					
52000B CM USED			22B	18					

SUBROUTINE INTERP 73 / 74 OPT#1

FTN 4.8+508

PAGE 1

```

1      SUBROUTINE INTERP (BR,BD,BH,RGRD,DGRD,HGT,IH1,IH2,PKS,PK,NR,ND,RU,012370
2      C DU,NH,NDBG) 012380
3      C
4      C   INTERPOLATES IN PK GRID TABLES. 012390
5      C
6      C   DIMENSION PKS(4C,20,8),RGRD(8,41),DGRD(8,21),HGT(9)
7      C   XINT(A,B,C,D,E) = E + (D-E)*(B-C)/(B-A) 012400
8      C
9      C   FOR EACH HEIGHT, FIND R,D BOUNDS WHICH BRACKET BURST 012430
10     C   POINT. 012440
11     C
12     C   P2 = -1. 012450
13     C
14     C   INITIAL PASS FOR LOWER HEIGHT BOUND. 012460
15     C
16     C   IH = IH1 012470
17     C   4 CALL FIND (BR,RGRD,NR,IH,IR1,IR2) 012480
18     C   CALL FIND (BD,DGRD,ND,IH,ID1,ID2) 012490
19     C
20     C   SET UP INTERPOLATION PARAMETERS & INTERPOLATE 012500
21     C   TO GET APPROXIMATE PK(FRAG). 012510
22     C
23     C   R1 = -RU + RGRD(IH,1) 012520
24     C   IF(IR1.NE.0) R1 = RGRD(IH,IR1) 012530
25     C   R2 = RU + RGRD(IH,NR) 012540
26     C   TF(IR2.NE.0) R2 = RGRD(IH,IR2) 012550
27     C   D1 = -DU + DGRD(IH,1) 012560
28     C   IF(ID1.NE.0) D1 = DGRD(IH,ID1) 012570
29     C   D2 = DU + DGRD(IH,ND) 012580
30     C   IF(ID2.NE.0) D2 = DGRD(IH,IC2) 012590
31     C   IF(BR.LT.R1.OR.BR.GT.R2) GO TO 7 012600
32     C   IF(BD.LT.D1.OR.BD.GT.D2) GO TO 7 012610
33     C   IF(NDBG.EQ.4) WRITE (6,*) *IR1,IH,IR2,IC2,D1,D2 012620
34     C   IR1,IR2,IC2,D1,D2,R1,R2,D1,D2 012630
35     C
36     C   INTERPOLATE FOR BURST RANGE ALONG LOWER DEFLECTION BOUND. 012640
37     C
38     C   PDI = 0. 012650
39     C   IF(ID2.EQ.0) GO TO 1 012660
40     C   PR1 = 0. 012670
41     C   IF(IR1.NE.0) PR1 = PKS(IR1,IR2,IH) 012680
42     C   PR2 = 0. 012690
43     C   IF(IR2.NE.0) PR2 = PKS(IR2,IR2,IH) 012700
44     C   PDI = XINT(R1,R2,PR1,PR2) 012710
45     C   IF(NDBG.EQ.4) WRITE (6,*) *PR1,PR2,IR1,IR2,PDI 012720
46     C   1 PDI = 0. 012730
47     C
48     C   INTERPOLATE FOR BURST RANGE ALONG UPPER DEFLECTION BOUND. 012740
49     C
50     C   IF(ID2.EQ.0) GC 1D 2 012750
51     C   PR1 = 0. 012760
52     C   IF(IR1.NE.0) PR1 = PKS(IR1,IR2,IH) 012770
53     C   PR2 = 0. 012780
54     C   IF(IR2.NE.0) PR2 = PKS(IR2,IR2,IH) 012790
55     C   PDI = XINT(R1,R2,PR1,PR2) 012800
56     C   IF(NDBG.EQ.4) WRITE (6,*) *PR1,PR2,IR1,IR2,PDI 012810
57     C   1 PDI = 0. 012820
58     C
59     C   INTERPOLATE FOR BURST RANGE ALONG UPPER DEFLECTION BOUND. 012830
60     C
61     C   IF(ID2.EQ.0) GC 1D 2 012850
62     C   PR1 = 0. 012860
63     C   IF(IR1.NE.0) PR1 = PKS(IR1,IR2,IH) 012870
64     C   PR2 = 0. 012880
65     C   IF(IR2.NE.0) PR2 = PKS(IR2,IR2,IH) 012890
66     C   PDI = XINT(R1,R2,PR1,PR2) 012900
67     C   IF(NDBG.EQ.4) WRITE (6,*) *PR1,PR2,IR1,IR2,PDI 012910
68     C   1 PDI = 0. 012920

```

SUBROUTINE INTERP 73/74 OPT=1 FTN 4.8+500 03/13/81 08.29.30 PAGE 2
 C INTERPOLATE FOR BURST DEFLECTION ALONG BURST RANGE, LOWER
 C HEIGHT.
 C 2 IF(IH.EQ.IH1) P1 = XINT(D1,D2,BD,PD1,PD2)
 C IF(NDBBG.EQ.4) WRITE (6,*),D1,D2,BD,P1
 C IF(IH1.EQ.IH2) GO TO 5
 C C INTERPOLATE FOR BURST DEFLECTION ALONG BURST RANGE, UPPER
 C HEIGHT.
 C IF(IH.EQ.IH2) P2 = XINT(D1,D2,BD,PD1,PD2)
 C IF(P2.NE.-1.) GO TO 3
 C IH = IH2
 C IF(IH2.EQ.0) GO TO 6
 C C REDO FOR UPPER HEIGHT BOUND.
 C GO TO 4
 C 6 P2 = 0.
 C IH2 = NH + 1
 C C INTERPOLATE FOR BURST HEIGHT.
 C 3 PK = XINT(HGT(IH1),HGT(IH2),BH,P1,P2)
 C RETURN
 C 5 PK = P1
 C RETURN
 C 7 PK = 0.
 C END

SYMBOLIC REFERENCE MAP (R=2)

ENTRY	POINTS	DEF LINE	REFERENCES	RELLOCATION	TYPE	SN
		1	82	84	F.P.	
321	D1	BD			REFS	0
322	D2	REAL			REFS	0
323	0	HGT	ARRAY	F.P.	DEFINED	0
315	ID1	REAL			REFS	0
316	ID2	REAL			REFS	0
312	1H	INTEGER			REFS	0

SYMBOLIC REFERENCE MAP (R=2)

ENTRY POINTS	DEF LINE	REFERENCES					
INTERP	1	82	84	86	88	62	68
VARIABLES	SN	TYPE	RELOCATION				
0 BD	REAL	F.P.	DEFINED	1			
0 BH	REAL	F.P.	REFS	61	DEFINED	1	
0 BR	REAL	F.P.	REFS	17	2*31	44	65
0 DGRD	REAL	ARRAY	DEFINED	1		45	
0 DU	REAL	F.P.	REFS	6	18	27	29
321 D1	REAL	F.P.	DEFINED	1			
322 D2	REAL	ARRAY	REFS	27	28	DEFINED	1
0 HGT	REAL	F.P.	REFS	52	33	61	63
315 ID1	INTEGER		DEFINED	27	28		
316 ID2	INTEGER		REFS	32	33	2*61	2*68
312 IH	INTEGER		DEFINED	29	30		
			REFS	6	3*81	DEFINED	1
			REFS	18	2*28	33	44
			REFS	18	2*30	33	50
			REFS	17	18	26	24

SUBROUTINE INTERP		73/74		CPT = 1	
VARIABLES	SN	TYPE	RELOCATION		
0 IH1		INTEGER		F.P.	
0 IH2		INTEGER		F.P.	
313 IR1		INTEGER		F.P.	
314 IR2		INTEGER		F.P.	
0 ND		INTEGER		F.P.	
0 NDBG		INTEGER		F.P.	
0 NH		INTEGER		F.P.	
0 NR		INTEGER		F.P.	
323 PD1		REAL		F.P.	
326 PD2		REAL		F.P.	
0 PK		REAL		F.P.	
0 PKS		REAL		F.P.	
324 PR1		REAL			
325 PR2		REAL			
327 P1		REAL			
311 P2		REAL			
0 RGRD		REAL			
0 RU		REAL			
317 R1		REAL			
320 R2		REAL			
FILE NAMES		MODE			
TAPE6		FREE			
EXTERNALS		TYPE	ARGS		
FIND			6		
INLINE FUNCTIONS		TYPE	ARGS		
XINT		REAL	5	\$F	
STATEMENT: LABELS					
125 1			46		
155 2			61		
213 3			81		
10 4			17		
225 5			83		
210 6			76		
227 7			85		

STATISTICS
BACCAN

二三八

LENGTH SEACON CROSSED

FTN 4.8+508	63/13/81	08.29.30	PAGE
16	70		
61	63	81	DEFINED 1
68	70	71	2*81
77			
2*24	33	2*41	2*52
2*26	33	2*43	2*54
29	DEFINED	1	
45	56	62	DEFINED 1
DEFINED	1		
25	DEFINED	1	
61	68	DEFINED	38 .44
2*61	2*68	DEFINED	46 .55
81	83	85	
41	43	52	54
45	55	56	DEFINED 40
45	2*55	56	DEFINED 42
81	83	DEFINED	61
2*81	DEFINED	12	76
17	25	24	26
25	DEFINED	1	
33	44	55	DEFINED 23
33	2*44	2*55	DEFINED 25
56		62	
61	68	81	

SUBROUTINE FIND 73/74 OPT=1

FTN 4.8+508

PAGE 1

```
1        SUBROUTINE FIND (B,GRD,N,IH,IX1,IX2)                          013230  
C                                                                                    013240  
C                                                                                    013250  
C                                                                                    013260  
C                                                                                    013270  
C                                                                                    013280  
C                                                                                    013290  
C                                                                                    013300  
C                                                                                    013310  
C                                                                                    013320  
C                                                                                    013330  
C                                                                                    013340  
C                                                                                    013350  
C                                                                                    013360  
C                                                                                    013370  
C                                                                                    013380  
C                                                                                    013390  
C                                                                                    013400  
C                                                                                    013410  
C                                                                                    013420  
C                                                                                    013430
```

SYMBOLIC REFERENCE MAP (R=2)

ENTRY POINTS	DEF LINE	REFERENCES						
3 FIND	1	15	18	21				
VARIABLES	SN	TYPE	RELOCATION					
0 B		REAL	F.P.	REFS	8	9	12	DEFINED 1
0 GRD		REAL	F.P.	REFS	7	8	9	DEFINED 1
37 I		INTEGER	ARRAY	REFS	11	12	12	DEFINED 10
0 IH		INTEGER	F.P.	REFS	8	9	12	DEFINED 10
0 IX1		INTEGER	F.P.	DEFINED	1	14	17	DEFINED 1
0 IX2		INTEGER	F.P.	REFS	14	1	20	DEFINED 1
0 N		INTEGER	F.P.	REFS	9	10	11	DEFINED 1
STATEMENT LABELS		DEF LINE	REFERENCES					
30 1		16	8					
33 2		19	9					
0 3		13	10					
25 4		14	12					
LOOPS	LABEL	INDEX	FROM-TO	LENGTH	PROPERTIES	INSTACK	EXITS	
16	3	1	10 13	7P				
STATISTICS	PROGRAM LENGTH	46B	38					
	52000B CM USED							

101 / 102

DISTRIBUTION LIST

Commander

U.S. Army Armament Research
and Development Command
ATTN: DRDAR-LCS, R. Corn
DRDAR-LCU, A. Moss
DRDAR-LCE-M, A. King
DRDAR-LCU-CA, D. Costa
DRDAR-LCS-A, J. Brooks (6)
DRDAR-LCS-E, S. Einbinder
DRDAR-TSS (5)
DRDAR-GCL
Dover, NJ 07801

Project Manager

Cannon Artillery Weapons System
ATTN: DRCPM-CAWS-El, G. Waldron
Dover, NJ 07801

Director

U.S. Army Materiel Systems
Analysis Activity
ATTN: DRXSY-GS, J. Chernick
DRXSY-MP
Aberdeen Proving Ground, MD 21005

Commandant

U.S. Army Field Artillery School
ATTN: ATSF-CD-AD, R. Wilde
Fort Sill, OK 73503

Director

U.S. Army TRASANA
ATTN: ATAA-TBA, J. Hodde
White Sands Missile Range, NM 88002

Commander

Naval Surface Weapons Center
ATTN: Code G29, R. Scheibe
Dahlgren, VA 22448

Commander

U.S. Army Electronics Research
and Development Command
ATTN: DELHD-R-RSD, R. Parkhurst
Adelphi, MD 20783

Commander

Naval Weapons Center
ATTN: Code 3524, J. Netzer
Chiua Lake, CA 93555

Administrator
Defense Technical Information Center
ATTN: Accessions Division (12)
Cameron Station
Alexandria, VA 22314

Commander/Director
Chemical Systems Laboratory
U.S. Army Armament Research
and Development Command
ATTN: DRDAR-CLJ-L
DRDAR-CLB-PA
APG, Edgewood Area, MD 21010

Director
Ballistics Research Laboratory
U.S. Army Armament Research
and Development Command
ATTN: DRDAR-TSB-S
Aberdeen Proving Ground, MD 21005

Chief
Benet Weapons Laboratory, LCWSL
U.S. Army Armament Research
and Development Command
ATTN: DRDAR-LCB-TL
Watervliet, NY 12189

Commander
U.S. Army Armament Materiel
Readiness Command
ATTN: DRSAR-LEP-L
Rock Island, IL 61299

Director
U.S. Army TRADOC Systems
Analysis Activity
ATTN: ATAA-SL
White Sands Missile Range, NM 88002